NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DISTANCE ESTIMATION USING HANDHELD DEVICES

by

Souhaiel Ben Tekaya

June 2013

Thesis Co-Advisors: Singh Gurminder
Edward Fisher

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704–0188) Washington, DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2013	**3. REPORT TYPE AND DATES COVERED** Master's Thesis 06–20–2011 06–21–2013	
4. TITLE AND SUBTITLE DISTANCE ESTIMATION USING HANDHELD DEVICES			**5. FUNDING NUMBERS**
6. AUTHOR(S) Souhaiel Ben Tekaya			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943–5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The different capabilities mobile devices can offer in the field of distance estimation for military applications are explored in this thesis. Of particular interest is the potential for using computer vision techniques to estimate distance in an operational military environment. The methods used for this investigation include a review of past literature on computer vision techniques in this domain, as well as an exploration of the different capabilities mobile devices offer in terms of sensors and networking.

We present two potential solutions. The first is a simulation of a distance estimation algorithm that gives the distance to the target using a pair of hyper stereo images. The second solution is a web-based mobile application prototype developed in HTML5. This prototype is intended for the use of untrained forward observers. It goes through the basic steps of a call for fire mission as required by a forward observer, with a focus on distance estimation.

14. SUBJECT TERMS Distance Estimation, Mobile Devices, Computer Vision, Depth Extraction, Stereoscopy			**15. NUMBER OF PAGES** 77
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540–01–280–5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DISTANCE ESTIMATION USING HANDHELD DEVICES

Souhaiel Ben Tekaya
First Lieutenant, Tunisian Army
B.S., Tunisian Military Academy, 2008

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL, AND COMMUNICATION)

From the

NAVAL POSTGRADUATE SCHOOL
June 2013

Author: Souhaiel Ben Tekaya

Approved by: Singh Gurminder
Thesis Co-Advisor

Edward Fisher
Thesis Co-Advisor

Dan Boger
Chair, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The different capabilities mobile devices can offer in the field of distance estimation for military applications are explored in this thesis. Of particular interest is the potential for using computer vision techniques to estimate distance in an operational military environment. The methods used for this investigation include a review of past literature on computer vision techniques in this domain, as well as an exploration of the different capabilities mobile devices offer in terms of sensors and networking.

We present two potential solutions. The first is a simulation of a distance estimation algorithm that gives the distance to the target using a pair of hyper stereo images. The second solution is a web-based mobile application prototype developed in HTML5. This prototype is intended for the use of untrained forward observers. It goes through the basic steps of a call for fire mission as required by a forward observer, with a focus on distance estimation.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
  A. OVERVIEW 1
  B. MOTIVATION 1
  C. PROBLEM STATEMENT 2
  D. MAIN CONTRIBUTION 2
  E. RELEVANCE TO DOD/DON 2
  F. METHODOLOGY 3
  G. ORGANIZATION OF THE THESIS 3

II. BACKGROUND 5
  A. METHODS FOR RANGE ESTIMATION 5
    1. Active Method: Emission of Electromagnetic (EM) Energy 5
      a. Advantages of Using LRFs 5
      b. Disadvantages of Using LRFs 6
      c. Advantages of Using Radar 6
      d. Disadvantages of Using Radar 6
    2. Passive Method: Computer Vision 6
      a. Advantages of Using CV 7
      b. Disadvantages of Using CV 7
  B. DEPTH EXTRACTION TECHNIQUES 8
    1. Monocular Depth Techniques 8
    2. Binocular /Multi View Techniques 9
  C. PLATFORM SPECIFICATIONS AND CONSTRAINTS 11
    1. Platform Description 11
    2. Mobile Sensors 11
  D. SUMMARY 13

III. DEPTH ESTIMATION MODEL 15
  A. MULTI-VIEW AND EPIPOLAR GEOMETRY 15
    1. Projective Geometry 15
    2. Camera Model 15
    3. Perspective Projection 16
    4. Definition of Epipolar Geometry 18
    5. Camera Parameters 19
  B. STEREOSCOPY 20
    1. Baseline 20
    2. Hyper-stereo 20
  C. DEPTH EXTRACTION ALGORITHM 21
    1. General Overview 21
    2. Image Rectification 22
    3. Disparity Map Calculation 22
    4. Depth Extraction 24
    5. Speed Up Robust Features (SURF) Detector 24

6. Sum of Absolute Difference (SAD) Algorithm 24
7. RANSAC Algorithm 25
D. IMPLEMENTATION 25
1. Model Description 25
2. Image Rectification 26
3. Disparity Map Generation 27
4. Range Map Generation 29
5. Constraints and Limitations 29
E. SIMULATION SETUP 29
F. RESULTS AND INTERPRETATION 30
1. First Round Results 31
2. Interpretation 31
3. Second Round Results 32
4. Interpretations 37
G. SUMMARY 38
IV. MAP-BASED RANGE ESTIMATION 39
A. INTRODUCTION 39
B. BACKGROUND 39
C. SCENARIO 39
D. TOOLS 40
1. HTML5 40
2. API Protocol 41
3. Webkit 41
E. MODEL PRESENTATION 42
1. Conceptual model 42
2. Operational Model 43
F. IMPLEMENTATION 49
G. TESTS AND RESULTS 49
H. CONCLUSION 50
V. CONCLUSIONS AND RECOMMENDATIONS 51
A. SUMMARY AND CONCLUSIONS 51
B. RECOMMENDATIONS AND FUTURE WORK 52
APPENDIX. SECOND EXPERIMENT OF DEPTH EXTRACTION MODEL 53
LIST OF REFERENCES 57
INITIAL DISTRIBUTION LIST 59

LIST OF FIGURES

Figure 1. Micrograph of a Hall sensor (from [17]) 13
Figure 2. Two parallel lines at infinity meet at the vanishing point. This case is not easily modeled by Euclidean geometry (from [1]). 16
Figure 3. Camera and image plane coordinate frame with rays intersecting in the camera center Oc. (from [1]) 17
Figure 4. Perspective projection. 18
Figure 5. Epipolar geometry (from [6])...19
Figure 6. Viewing acuity relative to hyper-stereo distance (from [4]).........................21
Figure 7. Rectification process diagram. 26
Figure 8. Disparity map generation process. 28
Figure 9. Lighthouse in the gray scale. 31
Figure 10. Depth map for the lighthouse target sample. 32
Figure 11. Motorcycle chosen as target for the second set of experiments......................33
Figure 12. Figure 1 Target's view in gray scale. 34
Figure 13. Disparity map after processing the basic block matching algorithm. 34
Figure 14. Range map generated based on the previous disparity map. 35
Figure 15. Superposition of the target's frame in both left and right images...................36
Figure 16. Disparity map of the target's frame. 36
Figure 17. Depth map within the target's frame. 37
Figure 18. The conceptual model describes the different elements involved in the application design and the organization of information. 43
Figure 19. Compass interface where the user has to point the device toward the target. 45
Figure 20. Map interface: contains circles separated by 500 meters each and all centered in the pin that marks the location of the device.................................47
Figure 21. Form interface where the UFO needs to provide information about himself, the CFD and the target. 48
Figure 22. Final fire mission script assistant. 49

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Camera parameters required for depth extraction. 19
Table 2. Motorola Atrix 4G camera's specifications. 29
Table 3. First set of measures..30

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ALS	Ambient Light Sensor
AMR	Anisotropic Magneto-Resistive
APIs	Application Programming Interfaces
A-GPS	Assisted GPS
BSI	Backside Illumination
CFF	Call for Fire
CCD	Charge-Coupled Device
COTS	Commercial-Off-The-Shelf
CV	Computer Vision
DOM	Document Object Model
EM	Electromagnetic
FDC	Fire Direction Center
GPS	Global Position System
GLONASS	Globalnaya Navigatsionnaya Sputnikovaya System
HMD	Helmet-Mounted Display
IPW	Imager Pixel Width
ICD	Inter-Camera Distance
LRF	Laser Range Finders
PCA	Principal Component Analysis
RANSAC	RANdom SAmple Consensus
RGB	Red Green Blue
SIFT	Scale-invariant feature transform
SURF	Speed up Robust Features
SAD	Sum of Absolute Distance
UFOs	Untrained Forward Observers
WHATWG	Web Hypertext Application Technology Working Group
WGS84	World Geodetic System
W3C	World Wide Web Consortium

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

All praise due to Allah the Most Gracious, the Most Merciful. Without his help and blessings I couldn't achieve this work.

I gratefully acknowledge my advisor Professor Singh Gurminder for his guidance, understanding, and thoughtful insight throughout this thesis. Many thanks go to my co-advisor Mr. Edward Fisher for his time and advices.

Special thanks also to Professor Mathias Kölsch for offering his knowledge and assistance to solve the issues I faced.

I would like to dedicate this work to my parents Said and Fatma, for their sacrifices throughout my educational career. Without their love and dedications, I could never be who I am.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

Military technology is known to be the leader in experimenting with new technologies in the operational domain. In fact, it's good practice to adjust the manner in which we use our military equipment by experimenting with different ways of dealing with routine military activities. Computer vision is one of the leading technologies for the military. It enables the performance of essential activities with minimal human involvement. In fact, computer vision techniques allow the extraction and understanding of information from row images. In this thesis, we implement computer vision algorithms with the goal of adapting it to military operations, in particular for estimating distances to targets, which is a common and routine requirement for combat.

The other facet of this research is to propose a tool to perform a call for fire using mobile device technologies. The current generation of commercial-off-the-shelf (COTS) devices appears to offer an attractive platform to explore. Developing a mobile application that assists untrained observers during a call for fire seems to be a promising solution.

B. MOTIVATION

Handheld devices have become ubiquitous and have increased tremendously in terms of incorporated features and computational capabilities. Both of these qualitative and quantitative improvements lead us to think of the possibilities for handheld devices to be used for some military activities. The current generation of COTS mobile phones comes pre-equipped with a number of different sensors including GPS, proximity, magnetic, image and audio sensors. In addition, the computation capability of these devices now matches that of the PC just a few years ago. These devices are powerful, lightweight, affordable and highly usable. As a result, these devices can be used for many non-traditional applications.

One of the areas in which COTS handheld devices maybe useful is for untrained observers who need the ability to make requests for indirect fire support, but these

observers may not be proficient in the proper methodology to make a call for fire (that is, the procedure used to request indirect fire). Developing an application that assists them in performing their mission using tools available within handheld devices seems to be beneficial.

C. PROBLEM STATEMENT

Target distance estimation is a difficult skillset for beginner trainees to master in the accomplishment of call for fire missions, because it requires the use of laser range finders (and the consequent need for training and maintenance). Our goal is to eliminate the need for specialized range finding instruments by providing the same capability on COTS mobile devices, thus eliminating the need for expensive, specialized equipment and reducing the number of devices that the soldiers have to carry.

D. MAIN CONTRIBUTION

Our research investigates the application of new techniques for untrained forward observers. Also, we test the algorithm of depth extraction from hyper stereo images using mobile devices. Moreover, this research introduces a prototype of a call for fire application. We consider it a first step toward developing a complete solution for call for fire missions.

E. RELEVANCE TO DOD/DON

Our goal from this research is to improve the capabilities of the military in terms of distance estimation and call for fire mission performance. We are proposing an alternative to the laser range finders currently in use. Even though the military range finders exhibit good performance, they are very expensive and require special training to operate. The proposed application can be developed on relatively inexpensive COTS mobile smartphones and adjusted to meet the operational needs of the military.

F. METHODOLOGY

This research contains two independent approaches: The first deals with using computer vision techniques in a Matlab environment. The second approach is a mobile device application developed in a HTML5/JavaScript language.

G. ORGANIZATION OF THE THESIS

This thesis consists of the following chapters:

Background information and a condensed literature review of the different methods for distance estimation and, in particular, of the different computer vision techniques for depth extraction are provided in Chapter II. Also, we give an overview of the state-of-the-art in the field of mobile devices.

A solution for extracting distance information from a pair of images is detailed in Chapter III. We present the tools and discuss the algorithms used to develop this solution.

In Chapter IV we introduce our prototype of the Map-based application that helps accomplish the call for a fire mission with a focus on estimating target distance.

Finally, a summary of the research results and recommendations for further research are presented in Chapter V.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

To get a clearer understanding of the technologies used for distance estimation, there must be an understanding of the implemented methods as well as the tools available to develop other methods. This chapter introduces several well-known methods used to estimate distances to targets. We also include a discussion of the multiple sensors that are embedded in most of the major mobile devices.

A. METHODS FOR RANGE ESTIMATION

The main focus of this thesis is to investigate effective methods used to estimate the range to distant targets. In order to make a useful comparison among the different methods, we include a critical presentation of the different types of distance estimation methods. This will include a discussion of both active methods and passive methods.

1. Active Method: Emission of Electromagnetic (EM) Energy

Electromagnetic energy is a form of radiation emitted by the source and could be absorbed by other particles. By active method, we mean the tools that emit any kind of energy that may be detected or intercepted.

We limit the discussion here to laser range finders (LRFs) and radar, which are two of the most common technologies that use active emissions. LRFs are widely and extensively used in commercial industry, law enforcement and military operations. The interest in these devices comes from their efficiency in estimating a very accurate distance to a distant target. Following is a discussion of the major advantages and disadvantages of this method.

a. Advantages of Using LRFs

LRFs present the advantage of being accurate enough for military operations such as artillery spotting and guidance. VECTOR 23, for instance, is the ultimate rangefinder from the VECTOR family of rangefinders built by Vectronix. It has

an accuracy of ±5m and can reach up to 24 km [1]. LRFs perform well throughout military operations and have become known for both accuracy and efficiency.

b. Disadvantages of Using LRFs

LRFs suffer from their relatively high cost, especially compared to simple optical methods and devices. Most of the laser range finders that meet the criteria/specifications for military missions are relatively expensive.

Another issue with the use of LRFs is their vulnerability to energy interception, as is true with any EM emissions. Since LRFs use an active method, they may reveal the device user's position very quickly. This may happen if the other party performs a frequency scan and the LRF user uses the device for a long enough time to be detected.

The operation of LRFs also requires specialized training, which restricts their utility to only those who have been properly trained.

c. Advantages of Using Radar

Radar is another active method which has been widely used to calculate the range of both small and large targets Radar is characterized by its long range and relatively fine accuracy (dependent upon purpose and design) and is implemented widely in almost all mobile platforms.

d. Disadvantages of Using Radar

As with LRFs radar energy can easily be intercepted, it puts users at risk. In addition, radars are susceptible to modern countermeasures, including jamming. Finally, radar systems have high power consumption; hence, they require a lot of energy and are not suitable for handheld devices.

2. Passive Method: Computer Vision

In passive methods, the device does not exhibit any form of emissions. Hence, the system using a passive method cannot be intercepted or detected. A camera is a good

example of the passive method since it absorbs light (energy) and does not emit any energy.

The computer vision (CV) technique is considered here as an alternative to the laser method described above. This technique provides some advantages and overcomes some of the disadvantages of active methods. CV looks very interesting in terms of cost, integration and stealth. For these reasons, it is currently the focus of much research. Significant work has been done to extract the depth (distance from object) from an image.

a. Advantages of Using CV

CV is simple, stealthy and cost efficient. Once correctly implemented and parameterized, CV is an automated process for both manned and unmanned systems. The user does not interfere with the processing of the image, adding to its ease and comfort of use. Because it is a passive technique, zero emission of EM energy, it is not vulnerable to interception by continuous frequency scanning. This passive feature enhances stealth within any platform that uses CV.

The use of cameras brings about a shift with military operations operating within a reasonable budget. Most military personnel can be equipped with handheld devices hosting a camera sensor. The use of this simple and ubiquitous camera technology in estimating a target’s distance would cost very little for hardware, personnel, or training (due to the common usage of and familiarity with the devices for civilian applications).

b. Disadvantages of Using CV

CV poses some challenges in terms of accuracy and complexity of software design and algorithms. CV is mainly used to calculate short ranges. Its applications are found in robots, unmanned systems, and similar platforms. The main objective of its implementation to date is to detect objects and avoid them. Hence, using CV for longer ranges is currently a challenge to be overcome. Added to this is the complex nature of the algorithm necessary to accurately process the sensed data. This is due to the different parameters that lend themselves to analysis in this thesis. These

parameters will be discussed in detail, but can be summarized as the distance between cameras, imager pixel width, focal length, search block window, image scaling, and disparity range.

Another challenge is the design of the application model and the operational model. In fact, CV techniques are, in general, based on epipolar geometry theories. These techniques are demanding in terms of design and operational models.

B. DEPTH EXTRACTION TECHNIQUES

Depth extraction techniques can be classified into monocular and binocular types. Monocular cues provide depth information when viewing a scene with one eye. Binocular methods provide depth information when viewing a scene with both eyes through exploitation of the differences between the perceived images when processed together.

1. Monocular Depth Techniques

Focus/defocus is a major technique used for monocular depth extraction. In [2], the author states that this technique is based on the blur, which is considered as the earliest method used to extract depth from simple images. There are two main approaches. The first consists of employing different images with several focus properties in order to capture the variation of blur in the image across the different images. Even though this method seems to be reliable and exhibits good depth estimation, it requires the use of different optical systems simultaneously which prevents it from being implemented in mobile device applications. The second approach is to extract blur from a single camera. This seems simple, but the problem is that the scenes captured using advanced cameras do not show background as out-of-focus regions [3]. In [4], there is application of the first approach on mobile phones cameras. It implements, tests and validates the process of fast auto-focus based on a set of pre-defined lens-position intervals. This technique permits the generation of a depth map of the scene. This technique has only been experimented at very short range, in the realm of centimeters, given the small focal length available in current technology mobile phones.

2. Binocular /Multi View Techniques

In stereo vision there are many algorithms that have been proposed. They follow different strategies [5]:

Feature based approach: In this approach, algorithms create correspondence between the images based on a number of extracted features. These algorithms extract features of interest from the images and match them. The method of feature extraction from the images differs from one algorithm to another; some algorithms use an edge pixels technique, also called edge delimited intervals, to extract features and compare between them in the images like in [6]. Other algorithms use line segmentation or curves. An advantage of this approach is that it gives relatively accurate information in less time and complexity. On the other hand, it delivers sparse depth information.

Area based approach: In this approach, depth maps are calculated by making a correlation of the gray levels of the image patches in the segments of the image in consideration. This technique assumes that these segments present some similarities [7]. This approach is adequate for relatively textured areas. It may prove poor at occlusion boundaries and within featureless regions [7].

Phase based approach: In this class of methods, algorithms are based on the Fourier phase information, which is a variant of the gradient-based optical flow method. It considers the difference between the left and right Fourier phase images. An implementation based on this approach in [8] uses phase similarities at multiple scales. Also, the method proposed uses oriented edges as features to be extracted, and that are using steerable filters. The sub-pixel accuracy in disparity is achieved by employing the phase difference between matched feature points.

Energy based Approach: This approach solves the correspondence problem using the energy minimization technique. In [9], the energy minimization technique is applied for scene reconstruction. Another method based on the energy approach is presented in [5]; it considers a weakly calibrated stereoscopic system, while only the fundamental matrix of the camera is known. Also, this method does not deal with any rectification process.

The approaches and methods presented above demonstrate reasonable results but they mostly test well for very short ranges that do not exceed a few feet. There is some work being done in terms of experimenting with computer vision algorithms and methods for use at longer ranges. The following is an overview of the major work being accomplished in this arena.

A novel model presented in [10] allows for the generation of a depth map using stereo image and hyperstereo image pairs. This approach considers that the distance between camera positions in stereo image acquisition affects the accuracy of depth perception. In fact, a normal separation of eyes (mean 6.3 cm) enables depth perception accuracy for just above 3m and vanishes totally after about 300 m. Accordingly the proposed model relies on widening the baseline, which is the distance between stereo camera positions while taking the picture. This methods exhibits reasonable results up to 20m and range information loses fidelity beyond 25 m [10].

Another concept mentioned in [11] uses a special configuration of cameras. There are two main camera configurations: parallel camera (most used) and toed-in camera. The latter configuration is exploited in the above mentioned paper. A larger baseline is implemented to offer a wider depth of field. This model was designed to provide long distance estimation application. Even though the implementation of this model is complicated and requires significant infrastructure to install the toed-in cameras precisely, the test results show that the depth map was not precise enough. It appears that this method would be effective for coarse segmentation of image elements.

Parallel camera techniques are generally known as stereoscopy. Hyperstereo is another term that defines stereoscopy with a large baseline. Hyperstereo depth perception is of growing interest and promises interesting applications and uses. An application of the hyperstereo depth perception, which is worth mentioning here, is the helmet-mounted display (HMD). The U.S. Marine corps is currently performing operational tests on the TopOwl HMDs manufactured by Thales Visionics. Also, a similar hyperstereo design is being considered for future U.S. Army aviation programs [12].The Thales design is described as two cameras installed on the sides of the helmet and offers depth information displayed on a transparent glass screen in front of the pilot. The position of

the cameras offers a larger baseline than the regular baseline of the pilot's eyes. The HMD models offered would provide helicopter pilots with expanded operational capabilities. The experimental results show that this techniques works well for distances of approximately 100 ft. (~30 m) , but for distances beyond 200 ft. (~60 m) the hyperstereo effect becomes moot.

C. PLATFORM SPECIFICATIONS AND CONSTRAINTS

Recalling our goal, we want to take advantage of mobile device capabilities to get an estimation of the range to a target. The tools are, therefore, restricted to the available sensors and network capabilities of these mobile devices.

The following is a discussion of the mobile device platforms which are examined and the different capabilities that are available.

1. Platform Description

In our research, we are focused on using COTS smartphones and tablets. Given their extended capabilities and affordable cost as well as their ubiquitous availability, these devices present an interesting option for our research. In fact, the growing sales of smart devices is interesting and worth mention. In [13], Gartner noted that 1.2 billion smart devices will be sold in 2013, compared to 821 million devices purchased in 2012 worldwide.

Our discussion here is based not only on the mobile devices themselves, but also on the global connectivity and capabilities that accompany them. We will be in need of the data connectivity in which the mobile devices are an end point; the device has to have access to data as well as to the Internet. The scheme of the design will be detailed further in the coming chapters.

2. Mobile Sensors

The current generation of COTS mobile devices is equipped with an increasing array of sensors. This fact has led us to think about the possible ways to exploit these sensors for a range of military activities. A typical mobile device has the following

sensors: Cameras, Ambient Light Sensor (ALS), Proximity Sensor, Global Position System (GPS), Accelerometer, Compass, Gyros, and Backside Illumination (BSI). Below, we describe briefly the set of sensors taken into consideration in our research:

Camera: The camera is one of the first sensors to be integrated within mobile devices. This integration inherited a lot of drawbacks in terms of camera capabilities compared to regular professional cameras. In fact, cell phone camera manufacturing has to meet a number of demanding requirements that are difficult to satisfy. This list includes high image quality, compact size, low energy consumption, fast response, etc. In general, the market offers mobile cameras that have mini lenses with a fixed focal length (usually 4 to 6 mm) and a fixed aperture (around f/2.8) [14]. The cutting edge technology in mobile camera sensors is currently provided by Nokia with PureView technology. The Nokia 808 PureView's camera offers a set of leading specifications, such as 8mm focal length, 41 Mb resolutions and an aperture of f/2.4 [15].

Global Positioning System (GPS): In the beginning, GPS was exclusively designed for military applications. Once made available to the public in the 1980s, it became the core of many commercial applications (navigation, mapping, etc.). In the mobile phone world, GPS is employed slightly differently. Most of the smartphones use Assisted GPS (A-GPS). This technology does the same work with the help of intermediate servers in case of disconnection with the main GPS satellites. Also, many smartphones support the GLONASS (Globalnaya Navigatsionnaya Sputnikovaya System) GPS system for navigation purposes. GLONASS is the Russian equivalent of the American GPS system.

Compass: Compasses are attracted to the earth's pole using magnets. For smartphone devices, it is not feasible to implant magnets, since they would interfere with the cellular signal connection. Today, AKM Semiconductors share 95 % of the mobile phone compass market [16]. The technology used in AKM mobile compasses relies on the Hall Effect. The Hall Effect consists of applying a magnetic field on an electric current flow. The magnetic field diverts the moving current charges and pushes them to one side of the conductor. This results in an induced voltage that can then be measured and used to get the strength of the magnetic field that caused the deviation. A compass

can be designed by using multiple sensors in different directions and a disk (magnetic concentrator) to bend the magnetic field line. The figure shows a micrograph of the AK8973 Hall sensor used in an iPhone 3GS.

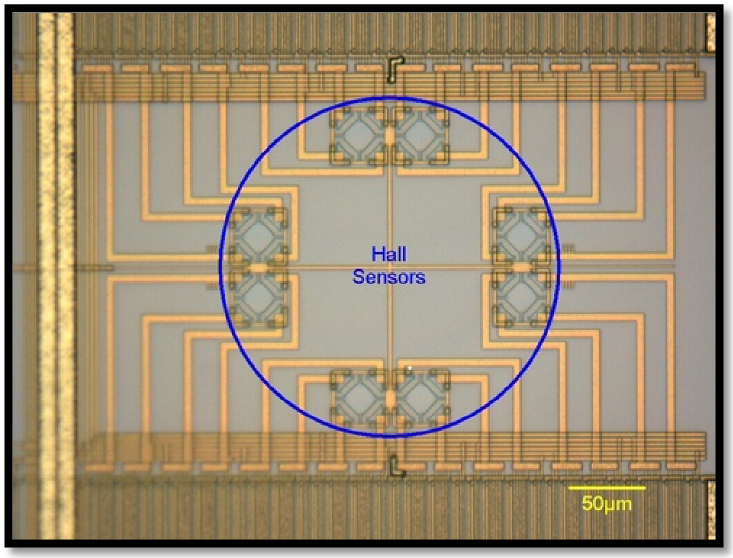


Figure 1. Micrograph of a Hall sensor (from [17])

Another type of sensor used as a magnetometer is the anisotropic magneto-resistive (AMR) sensor. It uses a thin film of ferromagnetic alloy that changes resistance according to an ambient magnetic field. They offer better accuracy, higher bandwidth and more temperature stability than Hall Effect sensors [18] .

D. SUMMARY

In this chapter, we have discussed several known methods for distance estimation. The active methods are presented and critiqued. We have also presented several different methods implemented in the passive method, particularly the computer vision techniques. Finally, we have presented different capabilities offered by the current mobile devices and their applicability in our research.

With this information as our background, in the following chapters we will investigate the implementation of CV techniques for long range estimation using the

integration of the sensors of interest available in mobile devices into a web-based mobile application to reach our accuracy goal.

III. DEPTH ESTIMATION MODEL

This chapter presents one of the solutions proposed for distance estimation using mobile devices. Starting with an introduction to the basics of the multi-view geometry necessary to understand the topic, we will then describe in detail the algorithm and techniques we implemented. Finally, we will discuss the experiments and interpret the results.

A. MULTI-VIEW AND EPIPOLAR GEOMETRY

Multi-view geometry, and in particular the epipolar geometry, is supported by the projective geometry. Epipolar geometry defines the geometry of stereo vision. In this section we define all the terms related to the perspective geometry as well as the parameters and definitions related to the camera.

1. Projective Geometry

Projective geometry represents the mathematical basis for the 3D multi-view imaging. It replaces the Euclidean geometry which presents a couple of disadvantages in the 3D space. The first of these is that points at infinity cannot be modeled and are considered as a special case [19]. An example of this issue is illustrated in figure 2.. Second, the projection of a 3D point onto a plane requires a perspective scaling operation (which itself requires a division that becomes a non-linear operation) [19].

2. Camera Model

From a computer vision stand point, and in order to process and interpret image information correctly, we need to choose the camera model that fits the project requirements. Several camera parameters should be identified and taken into consideration.

Figure 2. Two parallel lines at infinity meet at the vanishing point. This case is not easily modeled by Euclidean geometry (from [1]).

Since our research considers handheld devices, and given that cameras of these devices mostly use charge-coupled device (CCD)-like sensors, we chose to work with the pinhole camera model.

The pinhole camera model (also known as the perspective camera model) gives a description of the mathematical relationship between the coordinates of a point in the 3D space and its projection onto the image plane. The process is determined by choosing a camera center (that is, the optical center where all the rays intersect) and a projection plane. The projection process is as follows: all lines of light from the scene intersect in the center of the pinhole and are then projected inversely on the image plane. A pinhole camera model is shown in Figure 3. In an ideal pinhole camera model, the camera center is placed at the coordinate origin. The line from the camera center and perpendicular to the image plane is called principal axis, and the point P is the principal point. Also, in the illustration, point c represents the camera center while Z represents the principal axis. Both points x and p are located on the image plane. This geometric mapping is called perspective projection.

3. Perspective Projection

When viewing a scene, distant objects appear to be smaller than nearby objects. This is known as perspective. Perspective geometry is a description of the transformation of the scene from 3D to 2D world with the conservation of the objects' patterns (size,

distance, skew, etc.). Perspective projection of point *M* in the 3D world is described by these two equations where f denotes the focal length relative to the camera model:

$$x = \mathrm{f}\frac{X}{Z},$$
$$\mathrm{y} = \mathrm{f}\frac{\mathrm{Y}}{\mathrm{Z}}$$

Equation 1 Perspective projection equations

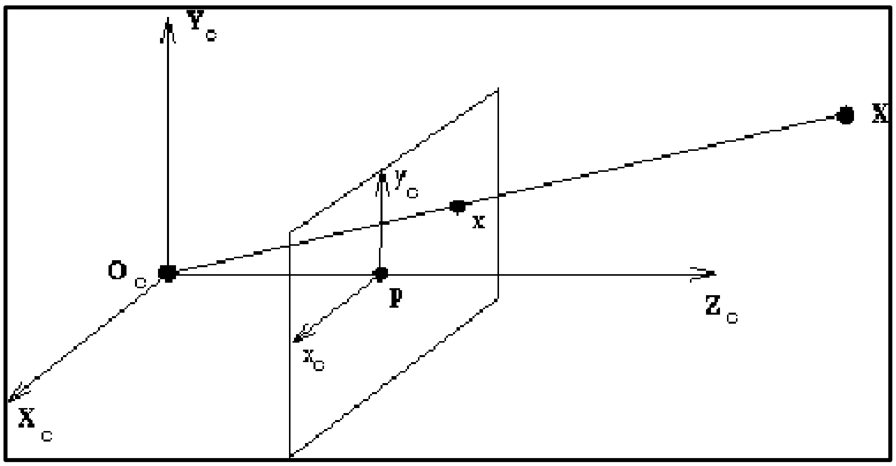


Figure 3. Camera and image plane coordinate frame with rays intersecting in the camera center Oc. (from [1])

In Figure 3, (x_c, y_c) represents the image frame (plane) coordinate system, while (X_c, Y_c) is the camera frame (plane) coordinate system. The above equations will work later as the basis for extraction of the depth map from the disparity map of the image.

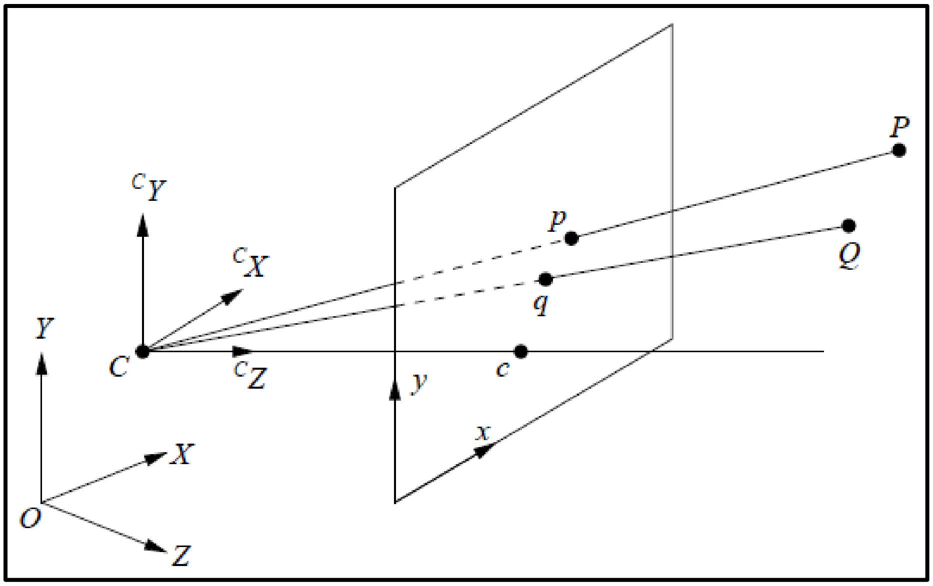


Figure 4. Perspective projection.

4. Definition of Epipolar Geometry

In general, the epipolar geometry represents the intrinsic projective geometry between two views [3]. It describes the relationship between the projections of an element in the real world in two image planes (frames) and is basically used for stereo imaging. In fact, epipolar geometry addresses two major aspects:

Point correspondence. From a point in one image plane, the epipolar geometry gives information on the position of the corresponding point on the other image plane.

Scene recovery. From the point correspondence and the camera position, epipolar geometry can reconstruct the scene structure.

This objective represents an element of the process that we will discuss to extract the depth from an image pair.

The illustration in Figure 5 represents the epipolar geometry of two cameras represented by their centers C and C' as well as the image planes. Image planes contain the 2D projection of the real world point X. These two projections as well as the point X lie on the common epipolar plane π. The baseline is the segment between C and C'. It

represents the separation between the two cameras while taking the image pair. The points where the baseline crosses the image planes are called the epipoles.

Epipolar geometry is independent from the scene structure, and depends on the camera's intrinsic parameters [3].

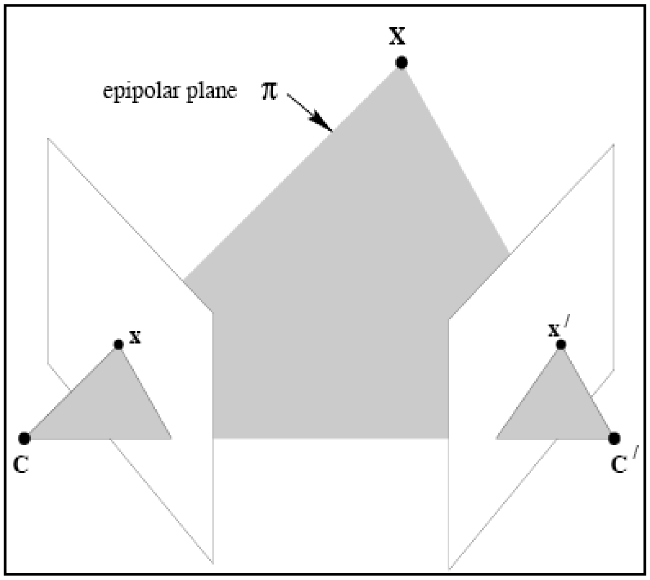


Figure 5. Epipolar geometry (from [6])

5. Camera Parameters

As mentioned above, the epipolar geometry (that will be the basis of the applied algorithm) depends on the parameters of the cameras. Hence, an accurate identification and description of the parameters of the mobile cameras to be involved in the testing is required. Table 1 depicts these different parameters.

Parameters	Description
Inter-camera distance (ICD)	The physical separation of the two mobile device cameras.
Focal length	The focal length of the camera in millimeters.
Imager Pixel Width (IPW)	The width of pixels on the camera sensor.

Table 1. Camera parameters required for depth extraction.

B. STEREOSCOPY

1. Baseline

In stereo photography, where the goal is to mimic natural human vision, the correct baseline (that is, the distance between where the left and right images are taken) to be considered is that of the distance between the left and right eyes of a human being. This distance is referred to as inter-pupillary distance (IPD). The mean human IPD is estimated to be equal to 6.3 cm [4]. We will test and decide upon the adequate baseline suggested to be used for a distant target's depth extraction.

2. Hyper-stereo

When a picture of a large distant object is taken using a normal baseline (between 5 and 8 cm), the object appears flat. In this condition, it is very difficult to extract any depth information related to the object.

An alternative is to increase the distance between the positions where both stereo images are taken. By doing so, the image will cover more of the scene, and we recover more details about it. Hence, more depth information will be available. This technique is known as hyper stereo imaging. In fact, according to [20], resolution in Z coordinates depends on the resolution of the camera used and the stereo base.

Another argument that supports the use of the hyper stereo technique for large distance estimation is shown in Figure 6. The chart describes the acuity of the results with regards to the variable baseline and for target distances of up to 70 feet (21 meters). As shown in the graph, by doubling the baseline, the acuity is improved remarkably. However, for baselines from three to eight times the original baseline, the acuity does not improve considerably beyond the previous results. This deduction is one of the reasons that led us to choose a baseline that is twice the IPD size for the second set of experiments. Other reasons for our choice are cited later in this chapter.

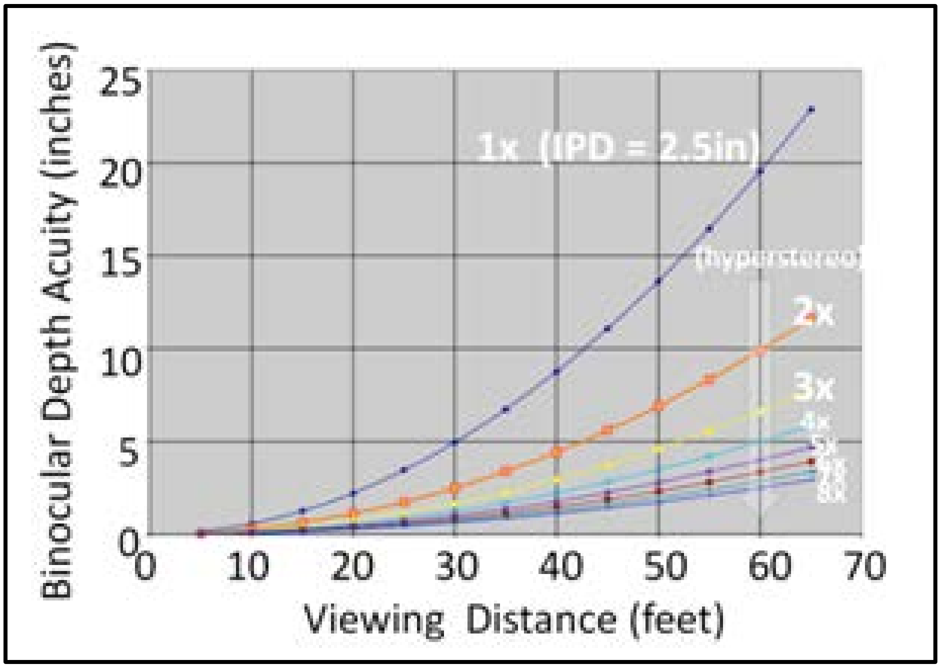


Figure 6. Viewing acuity relative to hyper-stereo distance (from [4]).

C. DEPTH EXTRACTION ALGORITHM

1. General Overview

The goal of this research is to investigate a method for an easy estimation of distant targets by using photo cameras. This challenge includes everything from image rectification up to disparity and depth map extraction.

In general, to proceed with the depth estimation algorithm, we have to make choices for each of the following three components [19]:

> **Matching criterion**. This criterion measures correlation or similarity between pixels. The choice here has no significant impact on the quality of the corresponding matching. In our case, we chose to work with the Sum of Absolute Distance (SAD) technique as described later in this section.
>
> **Support of the matching function**. This element describes the area upon which the matching function will be applied at one time. The algorithm in our case uses square windows.
>
> **Optimization strategy**. We will base our work on local optimization. This element calculates the disparity of each pixel using the single matching cost of the pixel independently. Local optimization yields accurate

estimation in textured regions. However, large texture-less regions produce fuzzy disparity estimates.

The following is a detailed description of the necessary steps to follow in the algorithm.

2. Image Rectification

Image rectification is the process of transforming two images such that their corresponding epipolar lines are horizontal and parallel [19]. It allows the projection of the pair of stereo images onto one common image plane. This process solves the correspondence problem that arises when taking a pair of stereo images using monocular cameras. In fact, image rectification simplifies the process of searching for corresponding points from the two-dimension to one-dimension level of complexity. After applying the process, pixels (in both images) relative to the same point in the real world are aligned at the same horizontal line.

3. Disparity Map Calculation

This step is crucial to extracting the depth information from the image. In this step, the algorithm goes over every pixel in one image and compares its position relative to its corresponding pixel in the other image. This process is very time and resource intensive. However, by achieving the previous step, the amount of computation is reduced to one dimension.

The general relationship between two corresponding points p_1 and p_2 can be written as follows:

$$\lambda_i p_i = K_i R_i \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - K_i R_i C_i \, , i \in (1,2)$$

Equation 2. Relationship between the two corresponding points p1 and p2 in the image plane.

where:

K: camera parameters matrix

R: camera rotation Matrix

λ: scaling parameter (aka homogenous scaling factor)

X, Y, Z: are the real 3D coordinates of the point P

p: point projection on the image plane (as shown in Figure 4)

Considering the case of rectified images and the pinhole camera model chosen, we have:

$C_1=0_3$, $R_1=R_2=I_{3x3}$, $C_2=(t_{x2}, 0, 0)$, where t_{x2} is the distance between the two camera centers (baseline).

Also, since the cameras are identical, we have:

$$K_2 = K_2 = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Equation 3. Camera parameters matrix

Equation 2 can now be written as follows:

$$\lambda_1 \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} = K \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

and,

$$\lambda_2 \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = K \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - K \begin{pmatrix} t_{x2} \\ 0 \\ 0 \end{pmatrix}$$

By combining both previous relations, it can be derived that:

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 - \dfrac{f.t_{x2}}{Z} \\ y_1 \end{pmatrix}$$

Equation 4. Relationship between corresponding pixels and depth

The quantity $\frac{f.t_{x2}}{Z}$ is called the disparity between the two pixels of the same 3D point in the image plane of each of the cameras. The disparity is inversely proportional to the depth.

4. Depth Extraction

This is the last step in the process of range estimation. It transforms the disparity information of the image into depth information based on the intrinsic parameters of the camera and other experimental parameters. The camera parameters are the focal length and the image pixel width. The experimental parameters are the inter-camera distance (ICD) and the disparity distance in pixels. This operation is based on Equation 4 about the relationship between disparity and depth information. The formula is reformulated in [4] by taking into consideration the parameters mentioned above and is described as follows:

$$z = \frac{f.ICD}{IPW.d}$$

Equation 5. Depth information extraction

5. Speed Up Robust Features (SURF) Detector

This algorithm is inspired by the Scale-invariant feature transform (SIFT) descriptor, but it is faster and more robust against images transformation than SIFT. In fact, the SIFT descriptor computes the derivatives over a patch of the image and reduces the large dimensional vector to a smaller one using the principal component analysis (PCA). By contrast, the SURF detector uses box filters to approximate the derivatives and integrals used in the first detector as mentioned in [21].

6. Sum of Absolute Difference (SAD) Algorithm

SAD is an algorithm that measures the similarities between blocks within two or more images. It calculates the absolute difference between every pixel in the original block and compares it to its corresponding pixel in the second block.

We mentioned earlier that the corresponding problem is reduced to the one dimension level using the rectification process. This dimension is the horizontal line of every image. Thus, for every block in the first image, its corresponding block lies on the same horizontal line on the second image. This line is called the epipole line.

We use the SAD algorithm for block matching for every pixel. The following equation is the SAD equation used for n x n windows with a disparity value (that will be introduced later): [22]

$$SAD(i,j,disp) = \sum_{h=1}^{n}\sum_{k=1}^{n}|LB(i+h,j+k) - RB(i+h,j+k+disp)|$$

Equation 6. SAD algorithm equation

7. RANSAC Algorithm

This algorithm was introduced by Fishler and Bolles in 1981. It is a robust estimator that is widely used in computer vision. It is described in [23] As both robust and simply implemented, and it performs well in problems where samples are contaminated with outliers. A particular use of this method is to detect outliers on the feature points. In this research we used RANSAC (RANdom SAmple Consensus) for this reason, as described in the model description shown in Figure 7.

D. IMPLEMENTATION

1. Model Description

The discussion of the depth extraction algorithm earlier in this chapter included the milestones of the entire process. A descriptive analysis of each of the components of this process is shown in Figure 7.

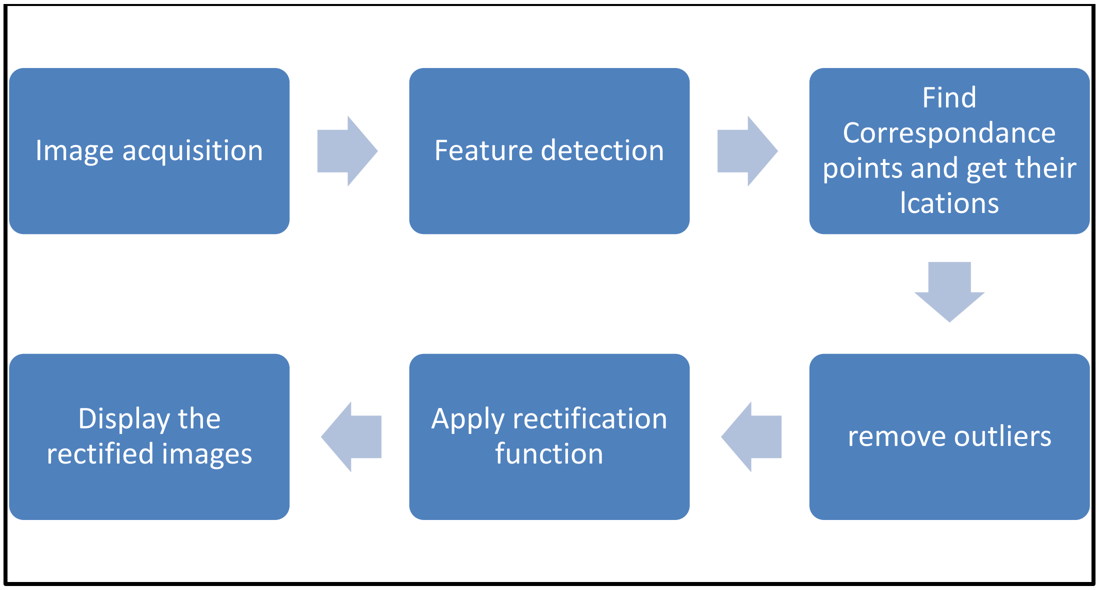


Figure 7. Rectification process diagram.

2. Image Rectification

The process of image rectification is described in Figure 7. Following is a detailed description of the algorithm.

Image acquisition. In this first part, we read both images and convert them from RGB (Red Green Blue) color space to gray scale. In gray scale, only the intensity of the pixels has value. According to [9], it is more efficient to work with only one-channel images, even though color images provide some improvement in accuracy. In fact, for this research we choose to compare the blocks of the images relative to their pixels intensity.

Features detection. After getting two images in the gray scale, we apply a feature extraction algorithm. For this work we chose the SURF descriptor. The function that has been used in Matlab is *detectSURFFeatures*. Based on the parameter *MetricThreshold*, we are able to manage how sensitive to features we want our algorithm to be. We leave this parameter to its default value of 1000. Another important parameter here is the number of features to keep and prepare for the following step. We tried different values before resolving this to 100. It appears that increasing the value beyond 100 may

significantly increase the computation time, while a lower number is not sufficient for the rectification process.

Finding correspondence and matching. After features have been extracted from every image, follows the process of matching them and estimating the corresponding feature in one image with its pair in the other one. The Matlab function *matchFeatures* uses the SAD method as a parameter and the two images' features as input. The output itself is the pairs of matched features and their indices in the images.

Removing outliers. The matched points are now used as an input for the Matlab function *estimateFundamentalMatrix.* Another required input for this step is the RANSAC method. This input allows excluding the outliers from the feature points. Another output from the considered function is the fundamental matrix. This contains the intrinsic parameters of the camera related to the stereo pictures. This variable will be used in the following step.

Applying rectification function. Once we get the inliers' corresponding points from both images and the fundamental matrix extracted, we input these variables into the *estimateUncalibratedrectification* Matlab function. Also, we apply projective rectification on the rectified images to present them on the same plane with their corresponding points.

3. Disparity Map Generation

This is the second step where we consider the rectified images and process them to extract the disparity map of the image pair. Here we intend to use the SAD algorithm for every pixel in the right image and search for the best matching in the same line (that is, the epipolar line) on the left image. A detailed description of the three elements involved in the disparity map generation process is shown in Figure 8.

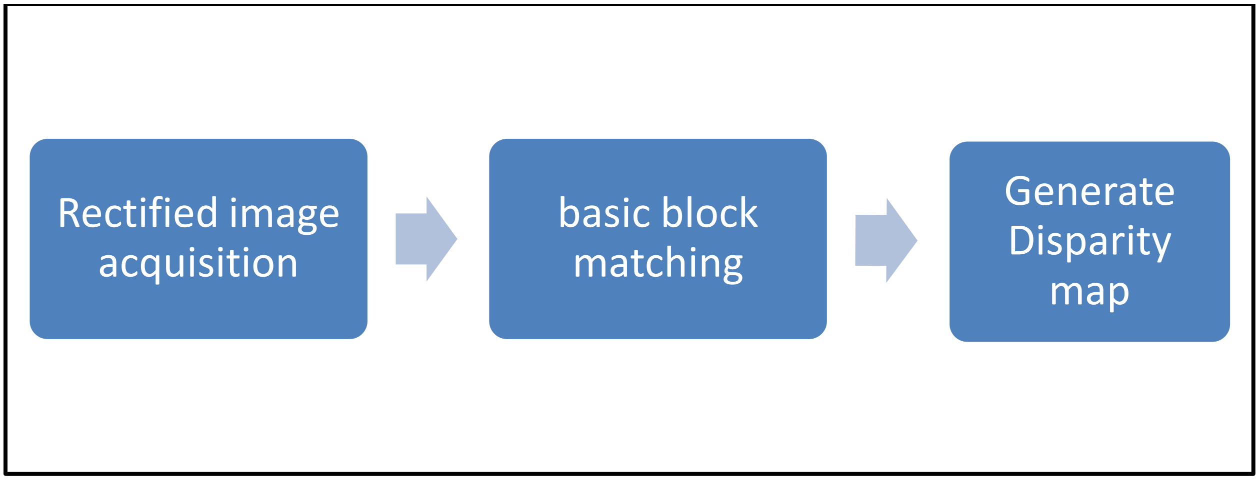


Figure 8. Disparity map generation process.

Rectified image acquisition. As we mentioned earlier, we should consider here only the rectified image generated from the previous process. In the Matlab environment, the rectified images are saved this time on matrix variable and not in an image file.

Another consideration here is that we cropped our region of interest from the image where the targeted object is located. This operation has the advantage of reducing the computing time required to go over the entire matrix (image).

Basic block matching. This is a critical step, where we have to define the parameters required for the SAD algorithm. These parameters are as follows.

> **Half block size.** This variable represents the half width of the rectangular block (matrix) to be applied around the pixel in consideration. We consider taking half the size to make its implementation easier in the equation.
>
> **Disparity range.** This variable defines how far the algorithm should consider searching for the pixel in the left image relative to its associate in the right image. There is an important relationship between the disparity range and the baseline. According to [24], a longer baseline results in a larger disparity range to be searched. This in turn implies that when working with a large baseline, we should increase the value of the disparity range.

In our work, we chose a block size of 7x7 and a value of 10 for the pixel disparity.

Generating disparity map. After the algorithm is applied over all the pixels in the left image, we can now generate the disparity map, which describes the displacement of an object in the real world relative to both images. The scale of the map goes from zero to the value assigned for disparity range (in our case ten).

4. Range Map Generation

This is the last step in the process of depth extraction. It is based on the equation mentioned earlier in "Disparity Map Generation." We apply that equation on every element of the disparity map of our region of interest. The equation requires some intrinsic parameters relative to the camera to be used for the operation. In the simulation setup we will introduce the camera used and its corresponding parameters.

After the range map is calculated for every element in the matrix, we take the average of the depth values to get an estimation of the target's distance vis-à-vis the camera.

5. Constraints and Limitations

Given that the algorithm was developed for stereoscopy imaging, and in order to get reasonable results, there are some constraints to consider for this work.

We assume that the target is immobile. The user should remember to take parallel images and simulate stereo cameras when taking pictures. The target location is given by the user. It is considered as an input in the algorithm.

E. SIMULATION SETUP

The image acquisition step is performed using a Motorola Atrix 4G device. This phone's camera has the characteristics described in the following table as in [25]:

Resolution	5 Megapixels
Focal length	4.34 mm
Pixel size	1.4 μm

Table 2. Motorola Atrix 4G camera's specifications.

The pixel size was not provided and difficult to determine, so we assumed its value to be the same as that of the iPhone 4S.

While acquiring the image pairs, we had to decide on the adequate baseline for our algorithm. The different measures taken for the first set of tests are summarized in Table 3.

Object	Real Distance (in meter)	Baseline (in meter)
Tree	175	10
House	182	10
Lighthouse	320	15

Table 3. First set of measures.

The ground truth distances to the target were taken using a golf range finder with a maximum range of around 800 feet (245 meters). Further distances are estimated using the Google Maps measuring tool. These experiments were conducted on two golf courses in Monterey, California.

The second set of images was basically an image pair that was taken according to the interpretations of the results of the first set. In fact, we reduced the baseline to approximately double the IPD (0.13 m), and the distance to the target was reduced to 88 m. Since the distance was small, we took the images within the NPS campus.

As mentioned earlier, this solution does not include the phase of detecting the target. Hence, the target is framed manually in the algorithm before proceeding to the disparity map generation.

F. RESULTS AND INTERPRETATION

This section presents the results of the experiments and the interpretations deduced from each one of them. The experiments were done in two rounds. We present both of them separately with their corresponding interpretations.

1. First Round Results

In the first round, we chose to work on a large baseline and a far distant target. A sample of the experiments done in the first set is shown in Figure 9. It shows the target lighthouse in the gray scale.

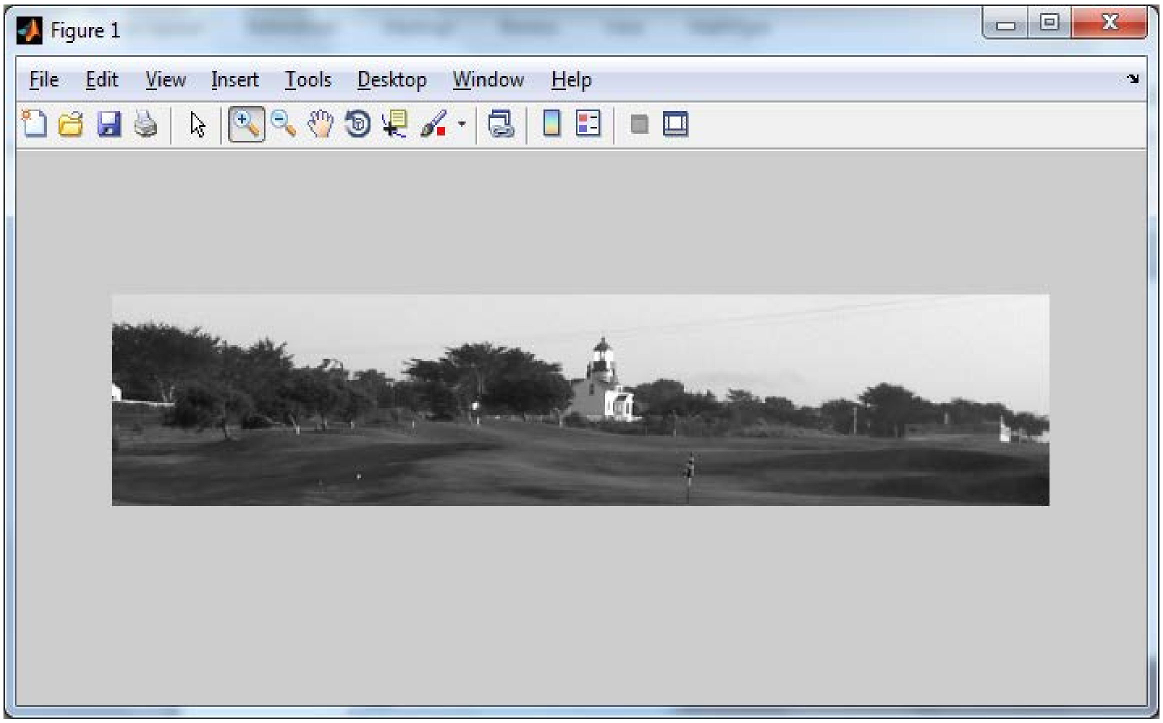


Figure 9. Lighthouse in the gray scale.

The depth map generated by the algorithm is presented in Figure 10. The depth map generation was not successful here since the information on the map does not correspond to the reality and did not correlate. The result was similar for the other targets.

2. Interpretation

We can interpret this result in different ways:

- The scene was featureless. The feature matching process has failed to collect enough features, and thus the rectification process would occur erroneously. As a consequence, the disparity map results in our failure.
- The baseline chosen was very large. As a consequence when shooting the picture the camera positions for the left and right images were far from

being parallel. In fact, the algorithm is sensitive to large vertical changes between the image pair.

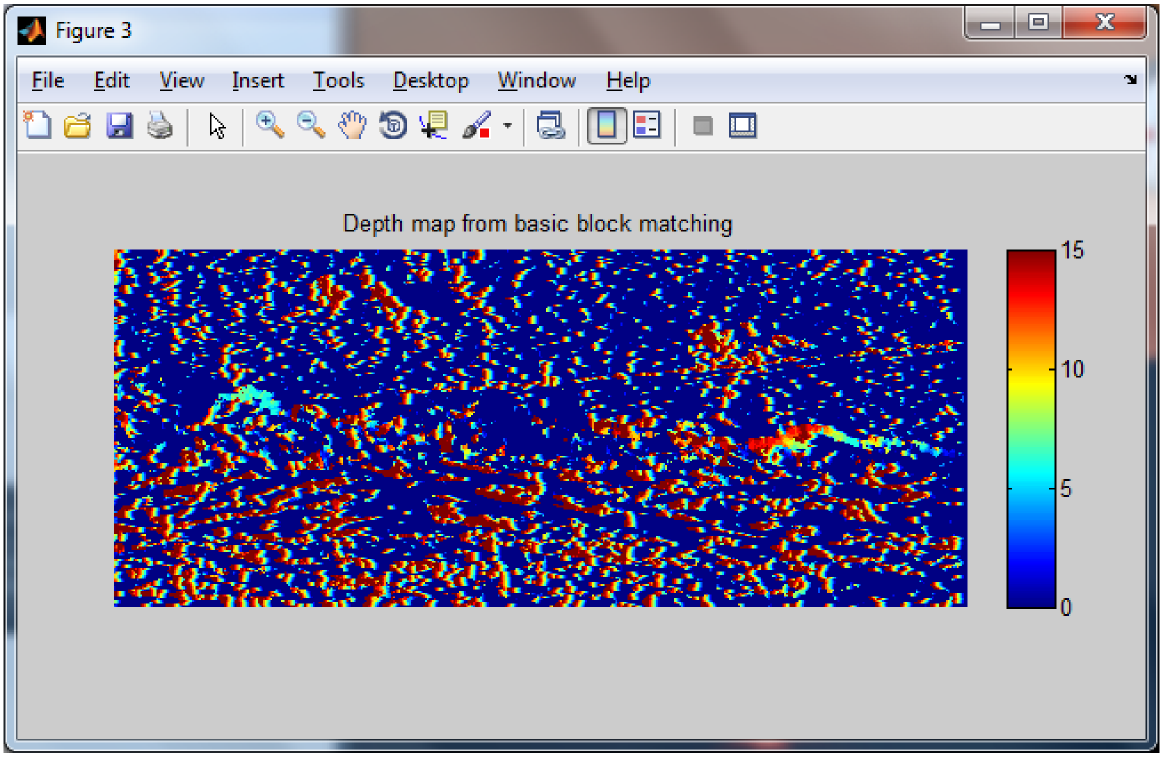


Figure 10. Depth map for the lighthouse target sample.

3. Second Round Results

The previous interpretations generated from Round 1 indicated a need to alter the experiments. The baseline was reduced to 0.13 m which represents twice the IPD. Also we used a closer target. A depiction of the left image relative to the target is shown in Figure 11. The yellow frame represents the inner matrix where we applied the algorithm.

Even though we only need the target to be processed, we had to select a smaller frame to validate the compatibility of the algorithm with the reality. In fact, the whole image is 2048 X 1536 pixels in resolution, meaning that it would take a significant amount of processing. The inner frame seemed more reasonable in terms of computation time and has quite enough depth space.

Figure 11. Motorcycle chosen as target for the second set of experiments.

The yellow outlined region (see Figure 11) is shown in gray scale in Figure 12. It will be useful later after the generation of the disparity map to decide the compatibility of the results compared to the ground truth. The resulting disparity map relative to the view is provided in Figure 13.

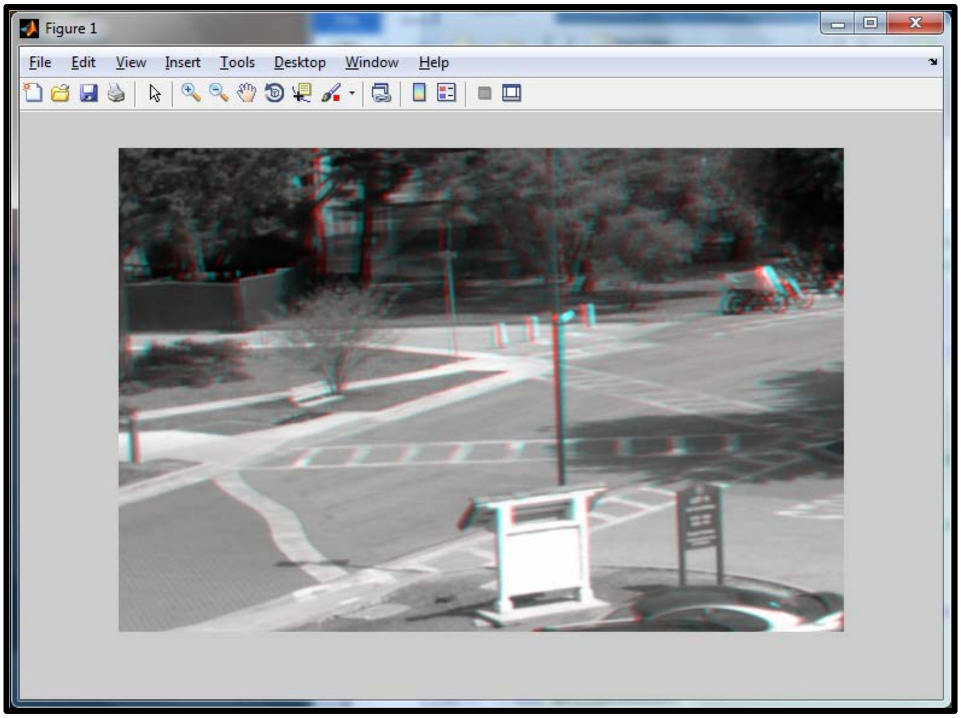


Figure 12. Figure 1 Target's view in gray scale.

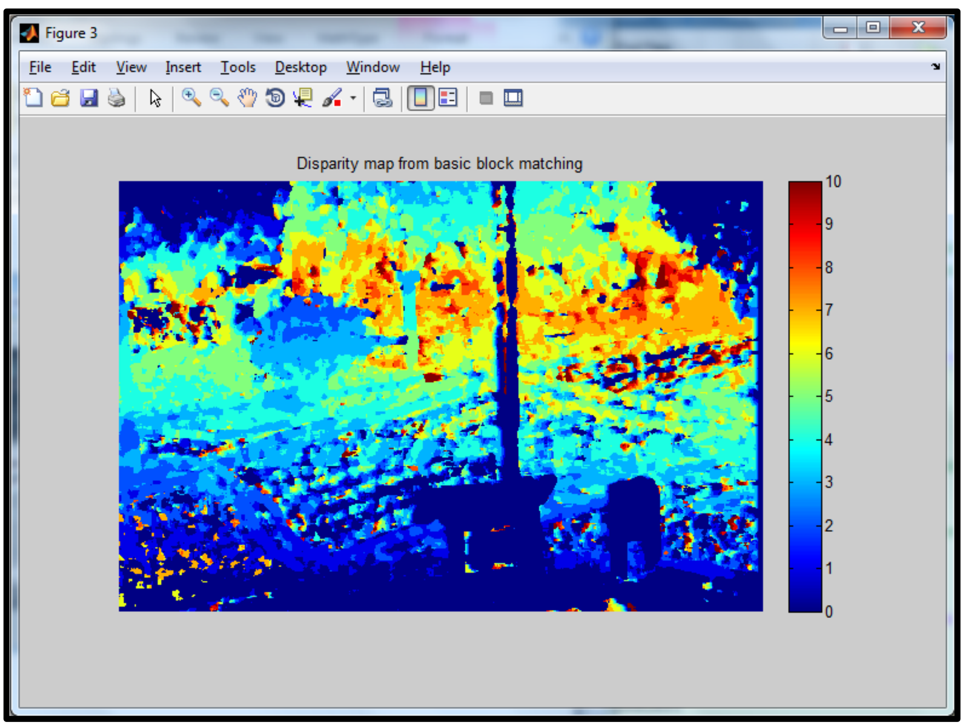


Figure 13. Disparity map after processing the basic block matching algorithm.

Moving forward to the next step (range map generation), we apply the algorithm to the disparity map. The results are illustrated in Figure 14 where the scale goes from zero to 300 meters.

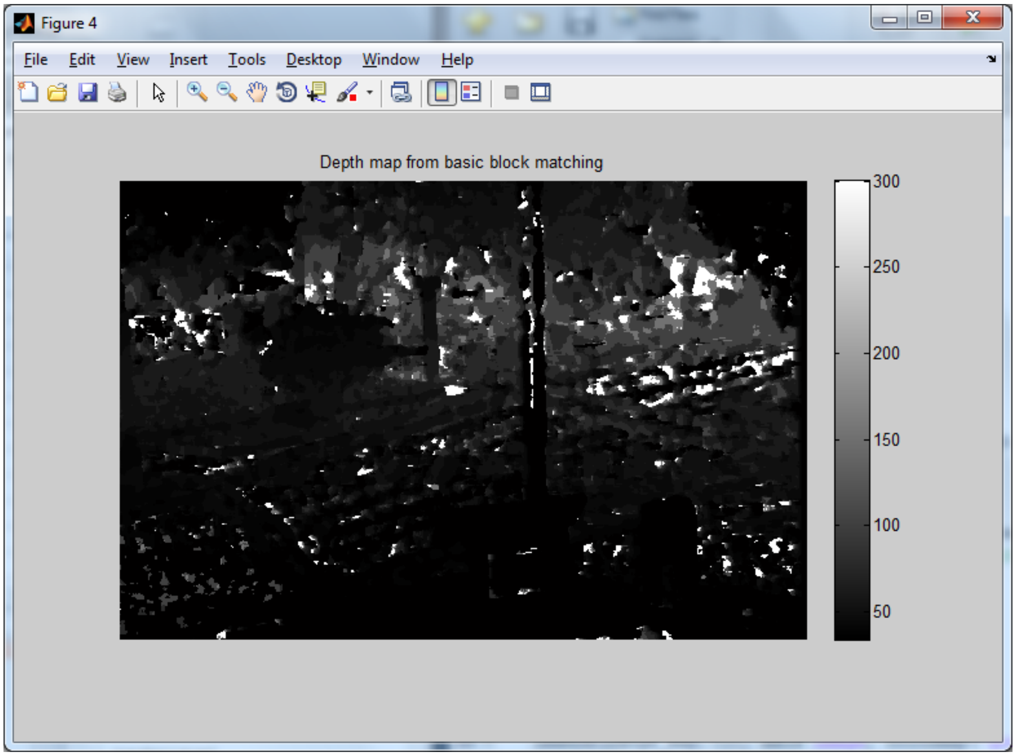


Figure 14. Range map generated based on the previous disparity map.

After running the algorithm in the neighborhood of the target and getting satisfactory results, we apply the technique on a smaller frame that contains the target. After choosing the target, a frame is created around it. This operation occurs for both images. The operation simulates the operator tapping on the target to select it. In Figure 15, one can see how the target's frames from both images are superposed. The image looks blurry because of the disparity between the left and right images.

The range map is calculated based on the disparity map and is shown in Figure 17. The output until this point is a matrix of points' depth. To get the depth information relative to the object being selected we take the mean of the values in the matrix. For the target (motorcycle) presented in the previous figures, we get a mean value of 24.28 m.

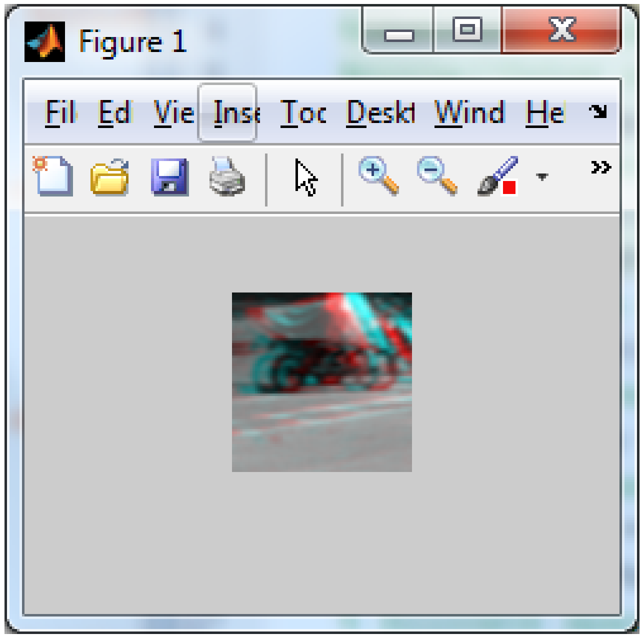


Figure 15. Superposition of the target's frame in both left and right images.

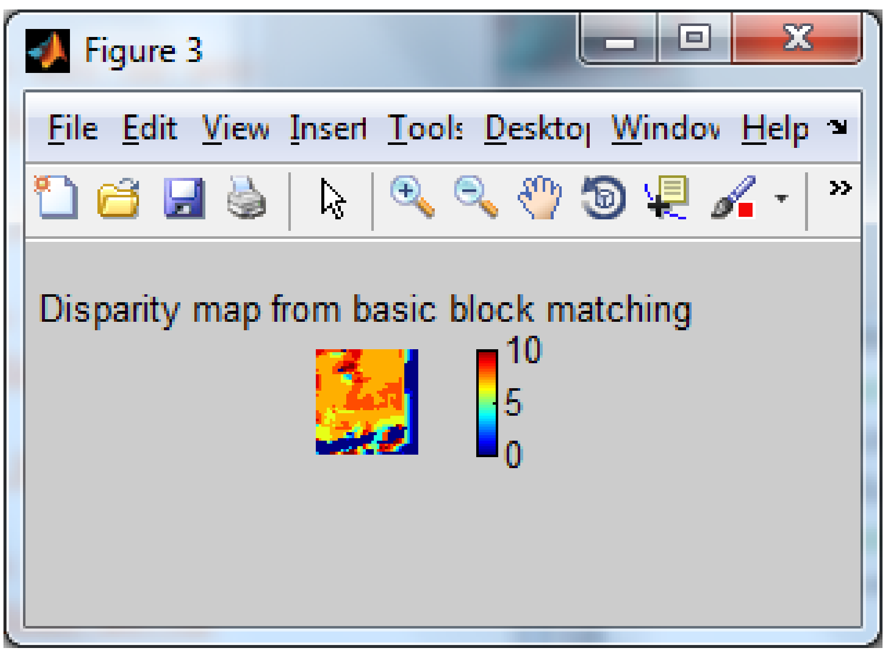


Figure 16. Disparity map of the target's frame.

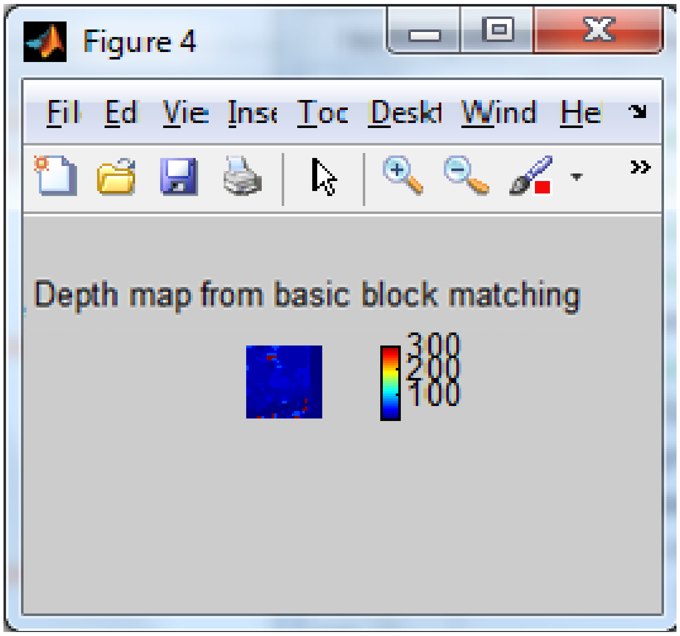


Figure 17. Depth map within the target's frame.

Another test with similar results was conducted and is illustrated in the Appendix.

4. Interpretations

Comparing both of the results depicted in Figure 11 and Figure 13, we can deduce that the output describes correctly the ground truth information: blue color pixels are the nearest, while reddish pixels correspond to the furthest objects. The results in Figure 14 are also acceptable since it represents a fair projection of the real depth of the scene.

Regarding the results relative to the target's frame, we notice that the output (mean value) is far below the ground truth value (around 88 meters). This might be interpreted as follows:

The frame chosen around the target contains some area that is closer to the camera than the target's position (lower area of the frame). This may affect the mean value of the frame.

The image pair was taken manually which led to some error regarding the ideal stereoscopy imagery. In other terms, the image pair did not represent a perfect stereo image pair. In fact, the algorithm adopted relies on stereo images.

For the present work, we worked with the SAD technique. This technique is sensitive to every pixel error. Added to this, this algorithm, as well the majority of depth extraction algorithms have good performance characteristics for short distances only, and this performance decreases considerably for longer distances.

Regarding the second experiment where the chosen target was a car (see Appendix), the output was much higher than ground truth (170 meters output compared to 70 meters in reality). This could be explained because there were some distant points in the background of the image.

G. SUMMARY

In this chapter we introduced the concept of computer vision and examined the depth extraction process in particular. We applied this technique in our research to measure how accurately we can recover distance information from hyper stereo images taken with mobile devices. We were able to extract depth information using SAD algorithm. However, it appears that the results are not accurate enough for military use.

IV. MAP-BASED RANGE ESTIMATION

A. INTRODUCTION

Recalling our objective in this research, we need to look for solutions that help estimate distances for distant targets. In this chapter we consider using the sensors and features of current mobile devices to build an application that helps produce this information and exhibits helpful results. To do so, and to make the solution more practical, we decided to take a case that is considered an important issue in the military and start from it to propose a solution. The issue is the need to improve the ability of Untrained Forward Observers (UFOs) to make a successful call for fire.

This chapter presents the solution from different aspects and is organized as follows: first, we introduce the background relative to the application as well as the technical tools and techniques applied to achieve the goal. We then describe our work from a conceptual point of view, and then the operational model describes the form of the application.

B. BACKGROUND

The present work is based on certain developmental techniques and platforms that need to be introduced in order to explain the topic. Also, we developed our application based on a specific scenario, which will be described to evaluate the application.

C. SCENARIO

The concept of using sensors embedded in handheld devices to facilitate call for fire missions is the focus of our solution presented here. We started from the assumption that handheld devices are ubiquitous, cheap and fully integrated with technology and sensors.

The idea is to explore their features and adopt them to the requirements of the mission. The proposed application is valid for training purposes and to assist a military in the field for untrained personnel with regards to how to place a call for fire.

Even though the scenario is described as a call for fire mission, we should specify that it deals particularly with a UFO.

D. TOOLS

In this section we present the techniques and standards used to develop the prototype. Basically, we used the HTML5 language associated with webkits relative to the browsers. Also, we used different Application Programming Interfaces (APIs) available for public use.

1. HTML5

HTML5 is a markup language introduced in cooperation with the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). It is a new standard for HTML. This standard is, however, not finished and work is still in progress, but the majority of web browsers support most of its API and elements. HTML5 is used to develop applications to run in most web browsers. The implementation of HTML5 (but not applications developed in HTML5) is browser dependent, unlike native programming languages (like Java) which are OS dependent. HTML5 introduces a number of new features and API:

Canvas. This allows for 2D drawing within a box of the web page.

Offline web applications. It’s the ability to use web applications offline, unlike typical online web applications that require a connection to the Internet to contact servers. This feature became available using two solutions: SQL-based database API (store data locally) and offline application HTTP cache (ensure availability of data while offline) [26].

Web storage. This feature behaves like cookies but with large storage capabilities.

HTML5 can access a mobile device’s sensors through the sensor API. They are defined by the W3C which supports DOM (document object model) events for device orientation, device motion, and whether the compass needs calibration [27].

2. API Protocol

Application programming interface (API) is a protocol used to permit software components to communicate with each other. Many APIs have been developed to standardize the process of dialogue between programs and servers or devices. Following is a discussion of the APIs that were used in our implementation:

Google Maps API: This API helps embed Google Maps into external sites or applications. Google Maps for Mobile works on Java based phones. One of the interesting features is the location detection. It uses GPS as well as wireless networks and cell sites for location data. The latter methods use triangulation as well as a database of known wireless networks. Google Maps API is free to use with web services that offer free access to consumers.

Geolocation API: This API provides a standardized scripted access to geographical location information associated with the hosting device [28]. It was introduced by the W3C. The most common sources for retrieving geolocation information are IP address, Wi-Fi, and GPS. The geographic position information is provided in terms of World Geodetic System (WGS84) Coordinates. Further details about this reference system are provided in [29].

Web storage API: This feature lets a web page store some information on the viewer's computer. It could be short duration or long lived. There are two types of web storage [30]:

Local storage: uses the *localStorage* object to store data permanently. Data will be on the user's system any time he/she visits the page.
Session storage: uses the *sessionStorage* to store data temporarily. Data remains until the user closes the window or ends the session.

In the current project, we used local storage since we do not have any session concept to use.

3. Webkit

Webkit is an open source web pages rendering engine software. This software can be used by almost any browser. In this prototype, we used the *transform webkit* feature

which allows making 2D transformations on objects. It was used particularly to rotate the compass objects.

E. MODEL PRESENTATION

As we mentioned earlier, the idea of developing the prototype came from the need to help a UFO user achieve his/her goal of calling for fire. We start by presenting the conceptual model which describes the flow of information and the steps required. Then, we present the operational model in which we introduce our vision of solving and implementing the solution.

1. Conceptual model

The conceptual model includes the flow of information along with the UFO scenario as well as the organization of the different elements that participate in it. A detailed description of the model in consideration is represented in Figure 18.

The system takes advantage of the sensors integrated in the mobile device to pull the necessary information for a call for fire briefing.

GPS: Extracts the position of the device

Compass: Gives the orientation of the horizontally positioned device relative to polar north.

Camera: Provides an image of the target before and after the shooting.

The user is additionally asked several questions in order to elaborate the brief at the end. Answers for this set of questions can be provided as a scroll down list from which the user can make a selection or answers can be typed by the user.

In addition to the above provided information, the application uses the Google Map API to import the map where the user is located, that location being provided through the Internet data that is offered by the carrier network.

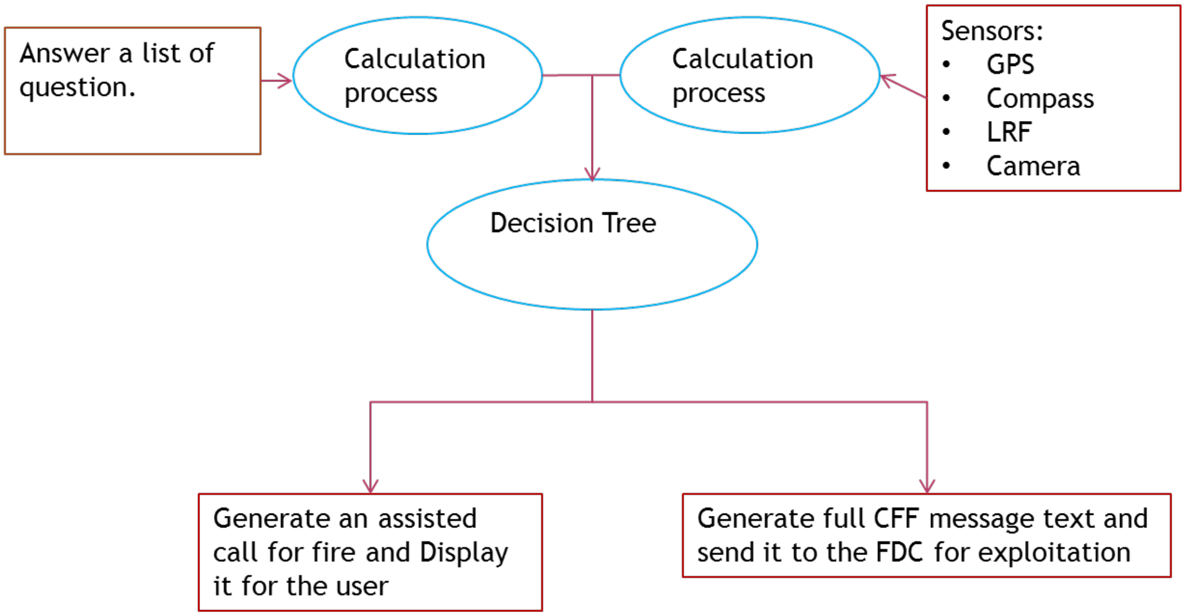


Figure 18. The conceptual model describes the different elements involved in the application design and the organization of information.

Once the system has acquired all of the necessary information, the information needs to be properly organized. Since we consider that the application targets the untrained forward observers, this means that the user requires full assistance all along the call for fire mission process. Hence, a standard model is designed, and will be generated and displayed at the end to help the UFO call the Fire Direction Center (FDC) and forward the Call for Fire (CFF) message. Another option to consider later is to generate the CFF message and send it automatically to the FDC.

After the information flow is described and the sources of information are identified, we detail later the operational structure of the application and the tools used to develop it.

2. Operational Model

The proposed solution is divided in two main parts. This covers the process of gathering information for the first part of the call for fire operation and does not cover the step of fire correction. However, the application can be extended later on since the

different elements that are presented in the following paragraphs are relatively independent.

Compass. This is the first element of the solution, where the user is invited to point the mobile device toward the target (see Figure 19). The device should be positioned horizontally. We used the *DeviceOrientation* event in JavaScript which polls the device for its orientation and makes it available for the application. *Transform Webkit* is used here to make the compass turn while the user turns the device to give him/her a feeling of a real compass.

Map. This is the second interface where the UFO is shown his/her position on the map with assistance to estimate the distance relative to the target. The map is downloaded using the data connection available through the main network. Google Map JavaScript API offers a set of parameters, among them we cite:

Center. Usually we define the device's coordinates to be the center of the map to be displayed.

Zoom. This parameter defines the initial resolution of the map when displayed.

Map type. The Google Maps API offers a set of map types for the user's needs. The different types are [31]:

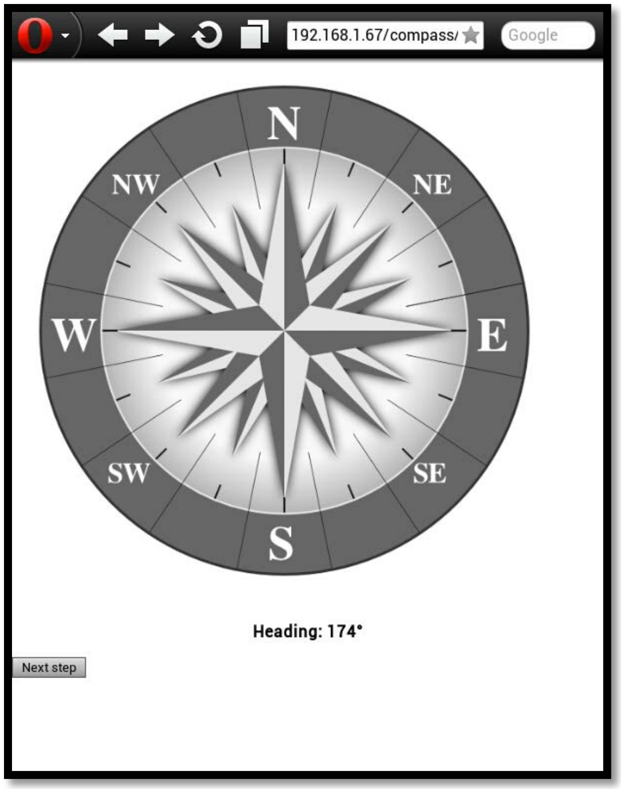


Figure 19. Compass interface where the user has to point the device toward the target.

Roadmap. Displays the default road map view.

Satellite. Displays Google Earth images.

Hybrid. Displays a mixture of normal and satellite views.

Terrain. Displays a physical map based on terrain information.

In the same interface, we used the geolocation API, which provides the longitude and latitude of the device's position.

Besides Google Maps and geolocation APIs, we used markers to assist the UFO estimate distance to any location visible on the map. Eventually, the scope of the area that can be accessed and estimated depends upon the zoom parameter that was set previously. This affects the accuracy and the visibility of small targets in the map. In the current prototype, we decided to draw three circles (markers) with a radius of 500, 1000, 1500

meters respectively. Hence, the UFO can make gross estimates of distance using circles separated by 500 meters.

Below the map, the user is invited to enter the direction measured in the previous interface as well as the distance to the target was estimated from the map using markers (circles) as described in Figure 20.

Form. This interface invites the UFO to answer a number of questions relative to his/her and the CFD's identifications as well as questions relative to the target as described in Figure 21. The list of questions includes the following:

What is your call sign? Identifies the UFO when launching a communication with the FDC.

What is your FDC call sign? Identifies the unit call sign to where the UFO refers when submitting the call for fire.

What is the mission type? In a regular call for fire mission, there are five types: Adjust fire, Fire for effect (standard), Suppression of Enemy Air Defense, Smoke, and Illuminate. In our prototype it should be put as standard.

What is the target? There are two ways to design how to answer to this question. It can be a scroll down list where the UFO chooses one element from the list, or we can keep the field empty and let the user identify what is observed exactly. We pick the latter choice.

How many are there? We need to know here the number of persons in the target location.

What is the target's degree of protection? The answer could be one of the following or a similar option: in the open, in trenches, overhead cover, in buildings.

Once the UFO is done providing all these elements of information, he/she saves them and continues to the last interface.

Fire mission script. This is the last interface in the prototype where the system assists the UFO in forwarding the message. Technically, the application pulls from the data stored in the local storage variables and organizes the data according to the standard of call for fire missions. This interface is designed so that the text to be read by the UFO is displayed in black while instructions are diplayed in red. The UFO has only to switch the radio on, select the correct frequency, read the text and follow the instructions. The message is forwarded. A sample of the script is displayed in Figure 22.

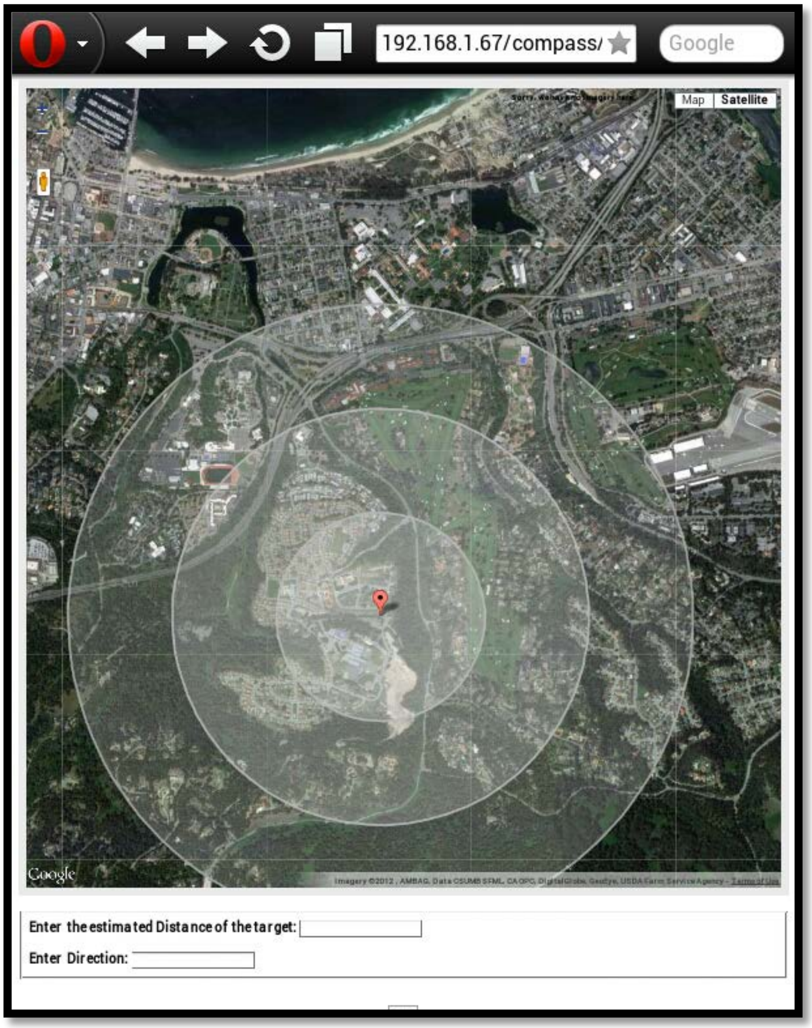


Figure 20. Map interface: contains circles separated by 500 meters each and all centered in the pin that marks the location of the device.

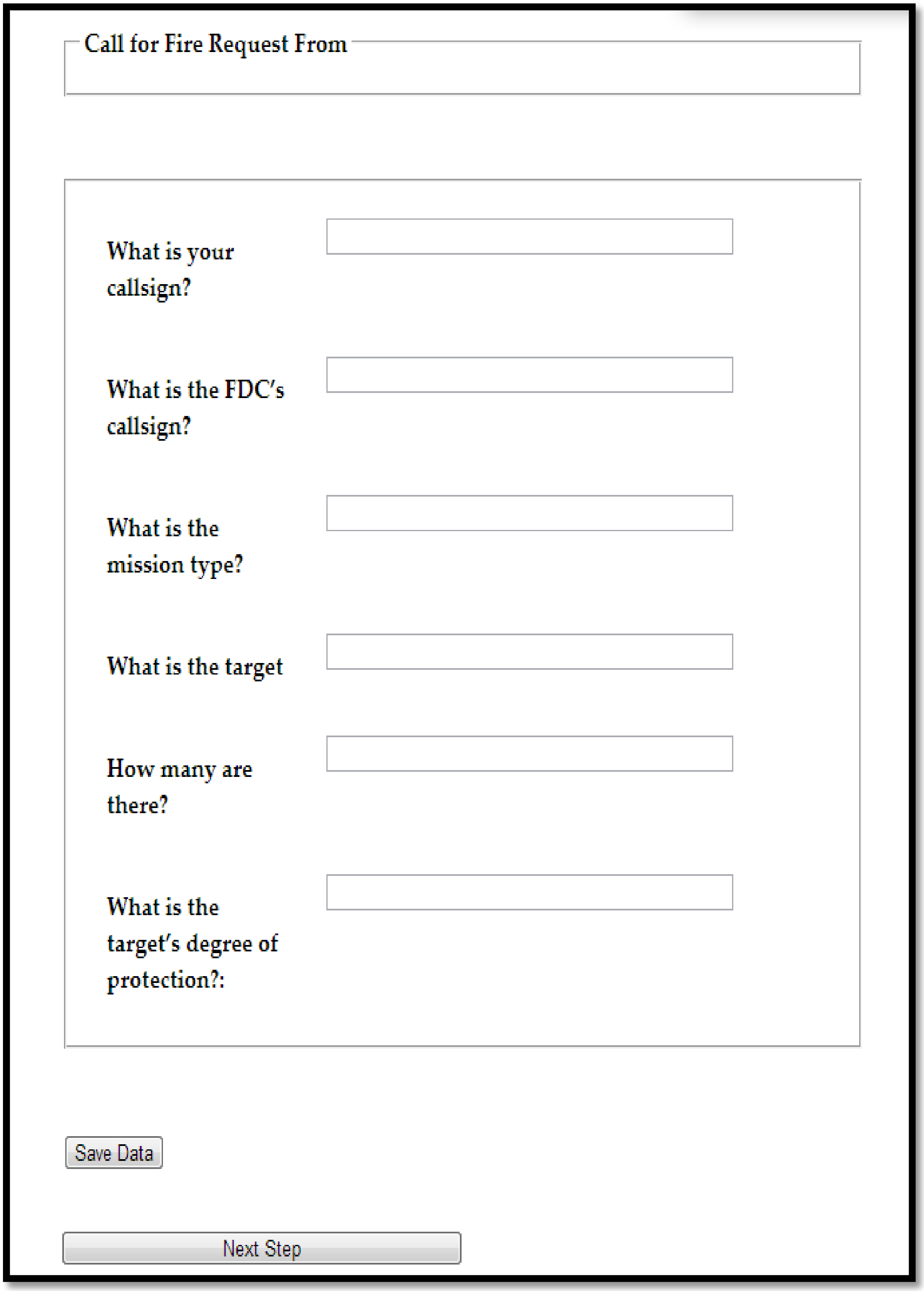


Figure 21. Form interface where the UFO needs to provide information about himself, the CFD and the target.

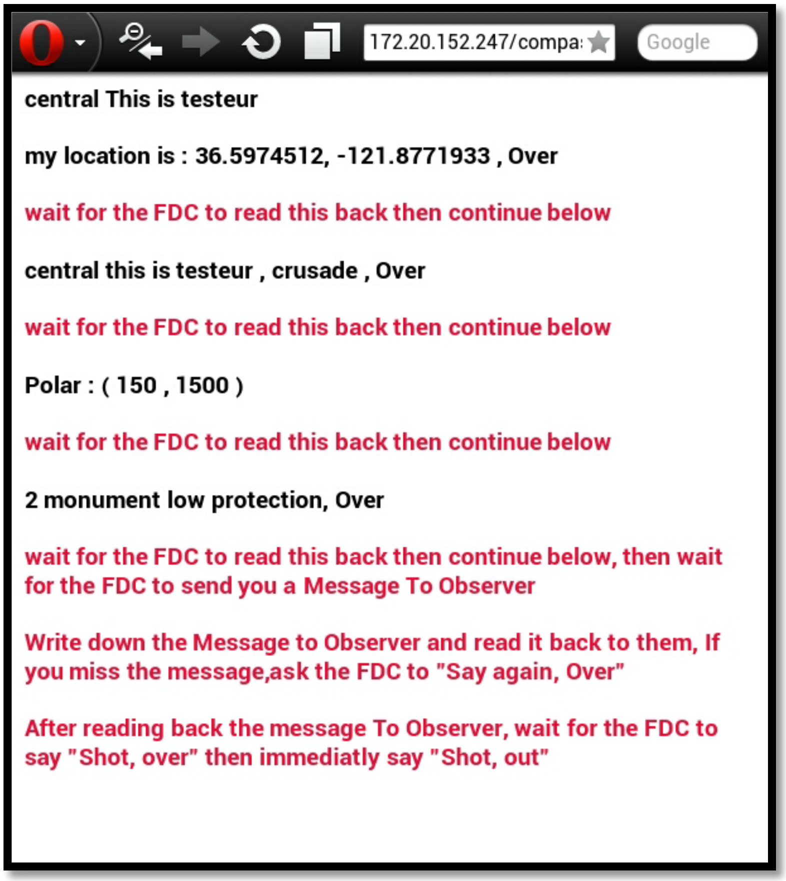


Figure 22. Final fire mission script assistant.

F. IMPLEMENTATION

The application was developed in HTML5 and with the various APIs described previously using notepad++ v6.3.2 editor and is hosted in a web server installed on a personal computer. The web server is Apache 2.4. The web server is contacted through the IP address that has been assigned to the personal computer in the wireless network.

G. TESTS AND RESULTS

The application was tested on several devices (iPhone, Android smartphone and Samsung tablet) and using different web browsers. As we mentioned earlier HTML5 is still a work in progress, and some browsers do not support certain webkits. In our case, *WebkitTransform* worked only in the Safari browser and is not supported in the Opera browser. To make the prototype universally applicable, we have to implement all the

webkit-equivalents for all the browsers. This means we need to implement the Gecko engine for Firefox, Trident for Internet Explorer, and Blink for Google browsers.

The compass direction is correct as long as the device is calibrated correctly, and the sensitivity of the device's movement is acceptable.

For direction we utilized units in both degrees and milliradians (military measurement unit), and it appears that it is more appropriate to work with degrees. The direction in degrees changes more slowly than the one in milliradians.

Regarding the map interface, where Google Map is displayed, we notice that the resolution of the map should depend on the target's distance, meaning that for a closer target we can increase the zoom to gain more accuracy when estimating distance. For longer distances, it would be better to decrease the zoom for the map to cover the target.

H. CONCLUSION

This chapter has introduced a solution to estimate distance using mobile devices. The solution was applied and adapted to call for fire missions. To develop this prototype, we used some new techniques such as the API and webkit along with the HTML5 standard for web applications. The results obtained after testing are positive and encourage expansion of the prototype to cover all the steps for a call for fire mission.

V. CONCLUSIONS AND RECOMMENDATIONS

The results reached in this thesis, as well as the limitations, problems, and challenges related to the research are summarized in this chapter. Moreover, this chapter highlights eventual avenues for future work.

A. SUMMARY AND CONCLUSIONS

This thesis was divided into two independent approaches. The first went through some of the computer vision algorithms and techniques for depth and disparity map extraction in the hyper stereo domain. To achieve our goal we had to implement a set of techniques. First we used the Speed Up Robust Features (SURF) descriptor to extract features from a pair of images. Second, we had to find the correspondence of these features and match them using the Sum of Absolute Difference (SAD) algorithm. Third, the RANSAC method was implemented to remove outliers from the set of feature points. The same SAD algorithm coupled with basic block matching technique was used to calculate disparities between corresponding points of the stereo images. Finally, the range map was extracted by applying the perspective projection equation on the disparity map already generated. The method used showed promising results but was still not precise enough to be implemented in the military domain.

The second approach applied in this thesis dealt with the same issue of distance estimation, but tackled it in a different way. We proposed another way to estimate distance using mobile device technologies and implemented it within an operational application. The application is a prototype of a call for fire assistance tool that is implemented on a mobile platform. It is oriented toward untrained observers but can be used by any operator easily. This application was developed in an HTML5 environment, and we used a number of APIs and webkits. Our purpose was to take advantage of the sensors available within mobile devices, such as compass, GPS and camera. HTML5 offers some features that enable web-based applications to continue to work offline with a database locally. Our overall idea is original, and, while we have not performed any

scientific testing, the prototype seems relatively successful in terms of ease of use and simplicity as well the accuracy of the results presented.

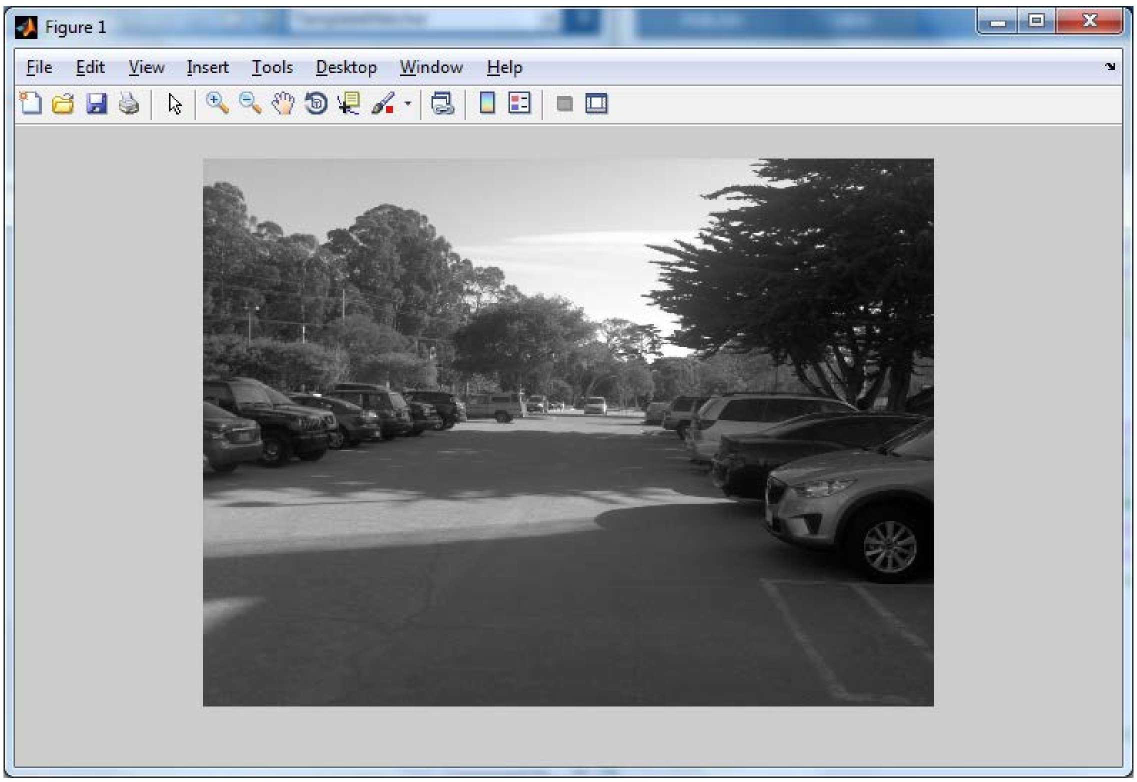
B. RECOMMENDATIONS AND FUTURE WORK

This thesis should be viewed as a first step toward developing a complete distance estimation tool and a call for fire training tool. Both approaches tackled in this research converge into one mobile application where we use the camera sensor to estimate distance using computer vision techniques. Hence, we suggest the following:

- Experiment with more computer vision algorithms and techniques to develop potentially more accurate results.
- Implement the algorithm used in the openCV (open computer vision) environment since it offers real-time implementation and has libraries that work better for mobile platforms.
- Continue development of the application in order to cover all the processes of a call for fire mission.
- Integrate the computer vision technique in a mobile environment.
- Test the application in various field exercises to collect usability and accuracy data.

APPENDIX. SECOND EXPERIMENT OF DEPTH EXTRACTION MODEL

This sample provided in this appendix illustrates the second experiment performed for the depth extraction model. This experiment was performed in a parking lot at the Naval Postgraduate School. The chosen target is a car at a distance of 70 meters. The following figures illustrate respectively the whole image in grey scale, the target in frames of both left and right images, the disparity map of the frame and the range map. The calculated mean of the final output is equal to 172 meters.



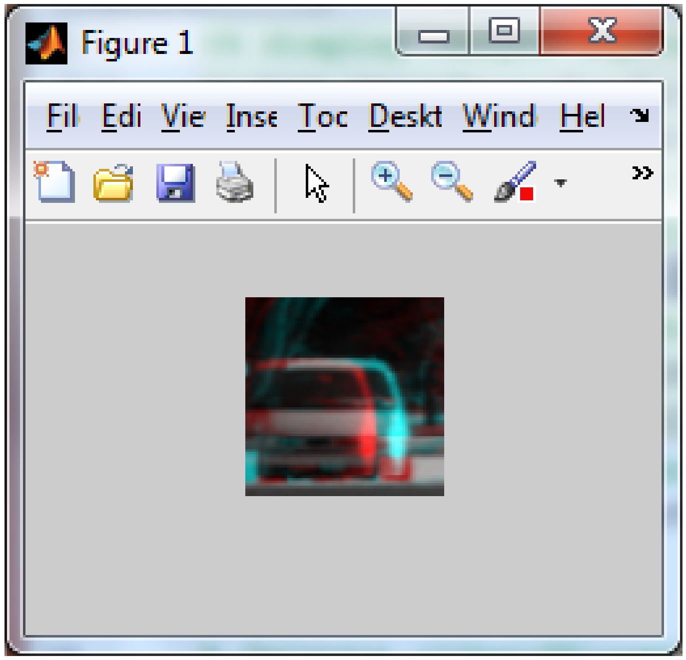


Figure 1
Fil Edi Vie Inse Toc Deskt Wind Hel

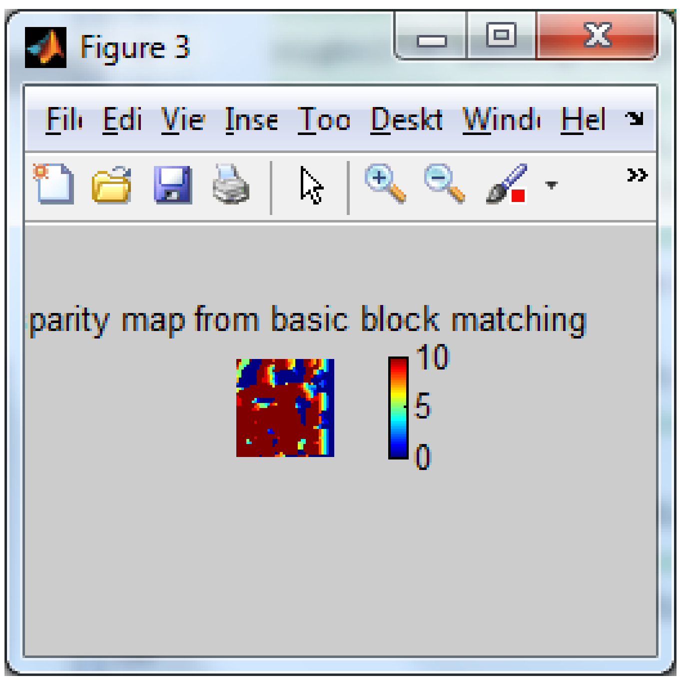


Figure 3
Fil Edi Vie Inse Too Deskt Wind Hel
parity map from basic block matching
10
5
0

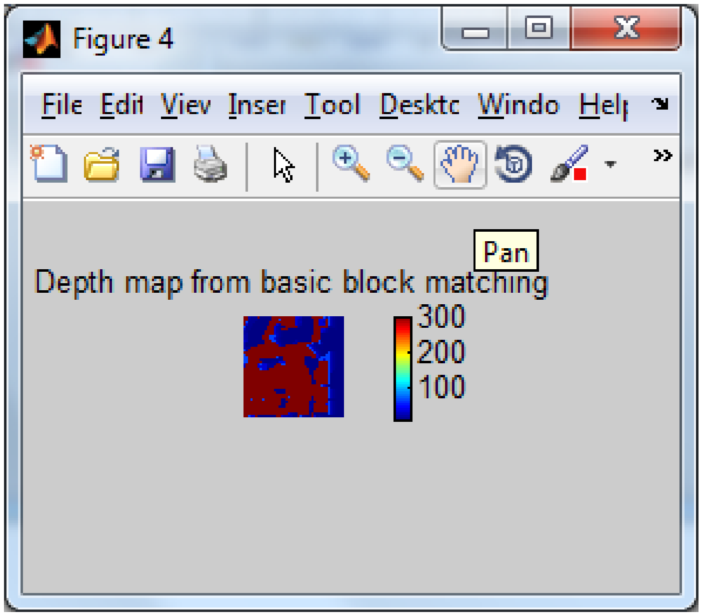


Figure 4
File Edit View Inser Tool Desktc Windo Help
Pan
Depth map from basic block matching
300
200
100

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] Y. Morvan, “Multi-View Depth Estimation” [Online]. Available: http://www.epixea.com/research/multi-view-coding-thesisch3.html#multi-view-coding-thesisse14.html.

[2] Subhashis Banerjee, “The pin-hole camera model” [Online]. Available: http://www.cse.iitd.ernet.in/~suban/vision/geometry/node3.html..

[3] R. H. a. A. Zisserman, *Multiple View Geometry*. Cambridge: University of Cambridge, 2003.

[4] Ashby, C., & Etienne-Cummings, R, “A novel 3D display using Multi-Hyperstereo Image Stitching,” in *Information Sciences and Systems (CISS),* 2011.

[5] K. O. a. K. Tomas, “Hyper stereo base camera setting for widescreen long distance target shooting application,” in *21st International Conference,* Brno, 2011.

[6] R. Szeliski, *Computer Vision: Algorithms and Applications*. London: Springer, 2010.

[7] X. Z. a. Z. Chen, “SAD-based stereo vision machine on a system-on-programmable-chip (SoPC),” *Sensors,* vol. 13, no. 3, pp. 3014–3027, 2013.

[8] K. T. a. Y. W. C. Sunglok, “Performance evaluation of RANSAC family,” in *British Machine Vision Conference (BMVC)*, 2009.

[9] Mathlab, Stereo Vision. [Online]. Available: http://www.mathworks.com/help/vision/ug/stereo-vision.html.

[10] O. Masatoshi and K. Takeo, “A multiple-baseline stereo,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 1993.

[11]] J. Sward, “Motorola Atrix camera specifications.” [Online]. Available: http://www.jeffreysward.com/editorials/atrixcam.htm..

[12] H. Cantzler, *Random Sample Concensus (RANSAC)*, Edinburgh: University of Edinburgh.

[13] Gartner, “Gartner says 821 million smart devices will be purchased worldwide in 2012; Sales to Rise to 1.2 Billion in 2013.” [Online]. Available: http://www.gartner.com/newsroom/id/2227215.

[14] R. Safaee-Rad; M. Aleksic, “Depth map from focus for cell-phone Cameras,” in Depth map from focus for cell-phone Cameras, San Jose, 2008.

[15] Nokia, Detailed specifications for the Nokia 808 PureView. [Online]. Available:http://www.nokia.com/global/products/phone/808pureview/specifications/#.

[16] Willie D. Jones, “A compass in every smartphone.” [Online]. Available: http://spectrum.ieee.org/semiconductors/devices/a-compass-in-every-smartphone.

[17] Elson Liu, How does the compass on the iPhone work? [Online]. Available: http://www.quora.com/iPhone/How-does-the-compass-on-the-iPhone-work.

[18] "Sensor platforms, understanding smart phone sensor performance: magnetometer understanding smart phone sensor performance: Magnetometer." [Online]. Available: http://www.sensorplatforms.com/understanding-smart-phone-sensor-performance-magnetometer-2/.

[19] Y. morvan, *Previous work on depth estimation*. [Online]. Available: http://www.epixea.com/research/multi-view-coding-thesisse14.html#x20-610003.3.1.

[20] K. Ondrej and K. Tomas, "Hyper stereo base cameras setting for widescreen long distance target shooting application," in Radioelektronika (RADIOELEKTRONIKA), 2011 21st International Conference, Brno, 2011.

[21] R. Szeliski, *Computer Vision: Algorithms and Applications*. London: Springer, 2010.

[22] X. Zhang and Z. Chen, "SAD-based stereo vision machine on a System-on-programmable-Chip (SoPC)," *Sensors*, vol. 13, no. 3, pp. 3014–3027, 2013.

[23] c. Sunglok, K. Taemin and Y. Wonpil, "Performance evaluation of RANSAC Family," in British Machine Vision Conference (BMVC), 2009.

[24] O. Masatoshi and k. Takeo, "A multiple-baseline stereo," IEEE transactions on pattern analysis and machine intelligence, vol. 15, no. 4, pp. 353–363, 1993.

[25] Jeffrey Sward, Motorola Atrix Camera Specifications. [Online]. Available: http://www.jeffreysward.com/editorials/atrixcam.htm.

[26] Anne van Kesteren, Offline Web Applications. [Online]. Available: http://www.w3.org/TR/offline-webapps/.

[27] Sensor platforms, "Sensors in HTML5." [Online]. Available: http://www.sensorplatforms.com/sensors-html5/

[28] Andrei Popescu, "Geolocation API Specification" [Online]. Available: http://www.w3.org/TR/geolocation-API/.

[29] National Imagery and Mapping Agency, "Department of Defense World Geodetic System," NIMA, 2000.

[30] M. Macdonald, *HTML5.* Sebastopol: O'REILLY, 2012.

[31] Google Developers, "Google Maps JavaScript API v3." [Online]. Available: https://developers.google.com/maps/documentation/javascript/maptypes.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California